

Phase 7A: Portfolio Analytics & Insights - Complete

Overview

Successfully implemented comprehensive portfolio analytics, enhanced custom collections, export history tracking, and trend visualization for the Genesis Provenance platform.

Status:  Production-Ready

Build: 66 routes, 0 TypeScript errors

Deployment: <https://genesisprovenance.abacusai.app>

Key Features Delivered

1. Portfolio Analytics Dashboard

A complete analytics page (/analytics) providing:

- **Real-time Portfolio Metrics:** Current value, total assets, verified/pending counts
- **Growth Insights:** Period-over-period growth rates, volatility analysis
- **Interactive Charts:** Value trends, asset count trends, category breakdown
- **Category Composition:** Detailed breakdown by category with percentages
- **Flexible Time Periods:** 7-day, 30-day, and 90-day views

2. Portfolio Value Tracking

- **Snapshot System:** Automated portfolio value snapshots for historical tracking
- **Trend Analysis:** Track portfolio value changes over time
- **Category Trends:** Monitor value trends by asset category
- **Statistical Insights:** Average value, highest/lowest values, volatility metrics

3. Enhanced Custom Collections

Upgraded saved searches with new capabilities:

- **Icon Support:** Assign custom icons to collections (e.g., "star", "bookmark", "heart")
- **Color Coding:** Add color codes for visual identification
- **Pinning:** Pin important collections to the top of the sidebar
- **Organization Sharing:** Share collections with entire organization
- **Smart Ordering:** Collections ordered by pinned → default → created date

4. Export History & Re-export

- **Export Tracking:** Automatically track all CSV/PDF exports
- **History View:** View past exports with metadata (date, file size, item count, filters)
- **Quick Re-export:** Re-generate previous exports with saved configurations
- **Automatic Cleanup:** Option to clear old export history (90+ days)
- **Filter Preservation:** Exports save applied filters for reproducibility

5. Interactive Visualizations

New `PortfolioTrendsChart` component with:

- **Area Charts**: Smooth portfolio value trends with gradient fills
- **Line Charts**: Multi-series asset count trends (total, verified, pending)
- **Stacked Bar Charts**: Category-wise value distribution over time
- **Responsive Design**: Adapts to all screen sizes
- **Interactive Tooltips**: Hover for detailed data points

File Structure

New Files Created

<code>/lib/portfolio-analytics.ts</code>	# Core analytics engine
<code>/app/api/analytics/portfolio/route.ts</code>	# Main analytics API
<code>/app/api/analytics/portfolio/snapshots/route.ts</code>	# Snapshot management
<code>/app/api/reports/history/route.ts</code>	# Export history list/cleanup
<code>/app/api/reports/history/[id]/route.ts</code>	# Re-export functionality
<code>/app/(dashboard)/analytics/page.tsx</code>	# Analytics dashboard UI
<code>/components/dashboard/portfolio-trends-chart.tsx</code>	# Chart component

Modified Files

<code>/prisma/schema.prisma</code>	# Added PortfolioSnapshot, ExportHistory, enhanced SavedSearch
<code>/app/api/saved-searches/route.ts</code>	# Added icon, color, pinning, sharing
<code>/app/api/saved-searches/[id]/route.ts</code>	# Updated schema validation
<code>/app/api/reports/export/route.ts</code>	# Added export history tracking
<code>/components/dashboard/dashboard-sidebar.tsx</code>	# Added Analytics link

Database Schema Updates

New Models

PortfolioSnapshot

```
model PortfolioSnapshot {}  
  id          String    @id @default(uuid())  
  organizationId String   @map("organization_id")  
  totalValue   Decimal  @map("total_value") @db.Decimal(12, 2)  
  totalItems   Int      @map("total_items")  
  verifiedItems Int     @map("verified_items")  
  pendingItems Int     @map("pending_items")  
  categoryBreakdown Json    @map("category_breakdown")  
  statusBreakdown Json    @map("status_breakdown")  
  snapshotDate  DateTime @map("snapshot_date")  
  createdAt    DateTime @default(now()) @map("created_at")  
  organization  Organization @relation(fields: [organizationId], references: [id], onDelete: Cascade)  
  
  @@index([organizationId])  
  @@index([snapshotDate])  
  @@map("portfolio_snapshots")  
}
```

ExportHistory

```
model ExportHistory {}  
  id          String    @id @default(uuid())  
  organizationId String   @map("organization_id")  
  userId      String   @map("user_id")  
  template    ExportTemplate  
  format      String  
  filters     Json?  
  fileName    String    @map("file_name")  
  fileSize    Int?     @map("file_size")  
  itemCount   Int      @map("item_count")  
  canReExport Boolean  @default(true) @map("can_re_export")  
  createdAt    DateTime @default(now()) @map("created_at")  
  organization Organization @relation(fields: [organizationId], references: [id], onDelete: Cascade)  
  user        User      @relation(fields: [userId], references: [id], onDelete: Cascade)  
  
  @@index([organizationId])  
  @@index([userId])  
  @@index([createdAt])  
  @@map("export_history")  
}
```

SavedSearch Enhancements

```
model SavedSearch {
    // ... existing fields ...
    isPinned      Boolean      @default(false) @map("is_pinned")
    isShared       Boolean      @default(false) @map("is_shared")
    icon          String?
    color          String?
    // ...
}
```

New Enum

```
enum ExportTemplate {
    full_csv
    summary_csv
    insurance_report
    tax_report
    appraisal_report
    custom
}
```



API Endpoints

Portfolio Analytics

GET /api/analytics/portfolio

- Query Parameters:
- days : Number of days for trends (default: 30)
- includeInsights : Include statistical insights (default: true)
- includeCategoryTrends : Include category-wise trends (default: true)
- Returns:

```
json
{
    "summary": {
        "totalValue": 1250000,
        "totalItems": 45,
        "verifiedItems": 38,
        "pendingItems": 7,
        "flaggedItems": 0,
        "categories": [
            {
                "categoryId": "...",
                "categoryName": "Watches",
                "totalValue": 450000,
                "itemCount": 15,
                "percentage": 36
            }
        ],
        "trends": [
    }}
```

```
{
    "date": "2024-11-15",
    "totalValue": 1200000,
    "totalItems": 43,
    "verifiedItems": 36,
    "pendingItems": 7,
    "growth": 2.5
},
],
"insights": {
    "currentValue": 1250000,
    "previousValue": 1220000,
    "growthRate": 2.46,
    "growthAmount": 30000,
    "highestValue": 1280000,
    "lowestValue": 1150000,
    "averageValue": 1215000,
    "volatility": 4.2
},
"categoryTrends": [...],
"periodDays": 30,
"generatedAt": "2024-12-01T12:00:00Z"
}
```

POST /api/analytics/portfolio

- Creates a new portfolio snapshot
- Returns snapshot ID and summary

GET /api/analytics/portfolio/snapshots

- Query Parameters:
- days : Historical period (default: 90)
- limit : Max snapshots to return (default: 100)
- Returns array of historical snapshots

Export History

GET /api/reports/history

- Query Parameters:
- limit : Max exports to return (default: 50)
- template : Filter by export template
- Returns list of past exports

DELETE /api/reports/history

- Clears exports older than 90 days

GET /api/reports/history/[id]

- Returns export configuration for re-export

DELETE /api/reports/history/[id]

- Deletes a specific export history entry

Enhanced Saved Searches

GET /api/saved-searches

- Query Parameters:
- `includeShared` : Include org-shared collections (default: true)
- Returns saved searches ordered by: pinned → default → created date

POST /api/saved-searches

- Body includes new fields: `isPinned`, `isShared`, `icon`, `color`

PATCH /api/saved-searches/[id]

- Supports updating all new collection fields



UI Components

PortfolioTrendsChart

Reusable chart component with three rendering modes:

1. Value Mode (`type="value"`):

- Area chart showing portfolio value over time
- Gradient fill and smooth curves
- Currency-formatted tooltips

2. Items Mode (`type="items"`):

- Multi-line chart for asset counts
- Separate lines for total, verified, pending
- Interactive legend

3. Category Mode (`type="category"`):

- Stacked bar chart by category
- Color-coded categories
- Shows value distribution over time

Props:

```
interface PortfolioTrendsChartProps {
  trends: PortfolioTrendData[];
  categoryTrends?: CategoryTrend[];
  type?: 'value' | 'items' | 'category';
}
```

Analytics Page

Full-featured analytics dashboard with:

- Period selector (7d, 30d, 90d)
- Refresh button for real-time updates
- Key metrics cards with growth indicators
- Tabbed chart views
- Category composition breakdown with progress bars



Portfolio Analytics Functions

Core Functions (lib/portfolio-analytics.ts)

`createPortfolioSnapshot(organizationId: string)`

- Calculates current portfolio metrics
- Creates snapshot record in database
- Stores category and status breakdowns as JSON

`getPortfolioTrends(organizationId: string, days: number)`

- Retrieves historical snapshots
- Calculates period-over-period growth
- Returns array of trend data points

`getCategoryTrends(organizationId: string, days: number)`

- Aggregates trends by category
- Returns time-series data for each category

`getPortfolioInsights(organizationId: string, days: number)`

- Calculates comprehensive statistics
- Includes growth rate, volatility, high/low values
- Returns financial insights

`getCurrentPortfolioSummary(organizationId: string)`

- Real-time portfolio summary without historical data
- Fast calculation for current state
- Used for dashboard cards

🎯 User Experience Enhancements

Before Phase 7A

- No historical portfolio tracking
- Basic saved searches without customization
- No export history or re-export capability
- Limited insights into portfolio performance

After Phase 7A

- **Portfolio Visibility:** Clear trends and growth metrics
- **Custom Collections:** Personalized organization with icons/colors
- **Export Management:** Track and reproduce exports easily
- **Data-Driven Decisions:** Statistical insights for portfolio optimization
- **Time-Series Analysis:** Understand value changes over time



🔒 Security & Performance

Security

- All analytics scoped to organization

- Export history user-specific
- Shared collections limited to same organization
- Authentication required for all endpoints

Performance

- Parallel data fetching with `Promise.all`
 - Indexed database queries on `organizationId` and `snapshotDate`
 - Efficient JSON storage for category/status breakdowns
 - Memoized chart data calculations
 - Responsive chart containers
-

Testing Guide

1. Portfolio Analytics Dashboard

```
# Navigate to analytics
http://localhost:3000/analytics

# Test different time periods
- Click 7d, 30d, 90d buttons
- Verify charts update
- Check metrics recalculate

# Test chart tabs
- Switch between Value Trends, Asset Counts, By Category
- Hover over data points for tooltips
- Verify responsive design on mobile
```

2. Create Portfolio Snapshots

```
# API call
curl -X POST http://localhost:3000/api/analytics/portfolio \
-H "Cookie: next-auth.session-token=..." \
-H "Content-Type: application/json"

# Expected response:
{
  "success": true,
  "snapshot": {
    "id": "...",
    "totalValue": 1250000,
    "totalItems": 45,
    "snapshotDate": "2024-12-01T12:00:00Z"
  }
}
```

3. Enhanced Collections

```
# Create collection with icon and pinning
curl -X POST http://localhost:3000/api/saved-searches \
-H "Cookie: next-auth.session-token=..." \
-H "Content-Type: application/json" \
-d '{
  "name": "High Value Watches",
  "description": "Premium timepieces over $50k",
  "filters": {...},
  "isPinned": true,
  "isShared": true,
  "icon": "star",
  "color": "#d4af37"
}'

# Verify in vault sidebar
# - Collection appears at top (pinned)
# - Icon and color displayed
# - Visible to all org members (shared)
```

4. Export History

```
# Generate export
curl -X GET "http://localhost:3000/api/reports/export?type=full&category=watches" \
-H "Cookie: next-auth.session-token=..."

# Check history
curl -X GET http://localhost:3000/api/reports/history \
-H "Cookie: next-auth.session-token=..."

# Expected: Export entry with metadata
{
  "exports": [
    {
      "id": "...",
      "template": "full_csv",
      "format": "csv",
      "fileName": "vault-export-2024-12-01.csv",
      "fileSize": 45230,
      "itemCount": 15,
      "filters": { "category": "watches" },
      "canReExport": true,
      "createdAt": "2024-12-01T12:00:00Z",
      "user": {...}
    }
  ]
}
```



Build & Deployment Status

Build Output

- 0 TypeScript errors
- 66 total routes (up from 64)
- Successful production build
- All new pages/APIs compiled

New Routes Added

- `/analytics` - Portfolio analytics dashboard
- `/api/analytics/portfolio` - Main analytics API
- `/api/analytics/portfolio/snapshots` - Snapshot management
- `/api/reports/history` - Export history
- `/api/reports/history/[id]` - Re-export

Deployment

Production URL: <https://genesisprovenance.abacusai.app>

Status: Ready for deployment



Future Enhancements (Phase 7B)

1. Predictive Analytics

- ML-powered asset value predictions
- Market trend integration
- Appreciation/depreciation forecasts

2. Advanced Visualizations

- Heat maps for category performance
- Correlation charts between categories
- Geographic distribution maps

3. Custom Reports

- User-defined report templates
- Scheduled automated reports
- Email delivery of insights

4. Benchmark Comparisons

- Compare portfolio to market indices
- Peer group analysis
- Performance percentiles

5. AI Insights

- Automated trend detection
- Anomaly alerts
- Optimization recommendations



Summary

Phase 7A successfully delivers:

- Complete portfolio analytics infrastructure
- Historical tracking with snapshot system
- Enhanced custom collections with icons, colors, pinning, sharing
- Export history and re-export functionality
- Interactive trend visualizations
- Statistical insights and growth metrics
- Production-ready build (0 errors, 66 routes)

Next Steps:

1. Deploy to production: <https://genesisprovenance.abacusai.app>
 2. Monitor analytics usage and performance
 3. Gather user feedback for Phase 7B enhancements
 4. Consider adding scheduled snapshot creation
-

Phase 7A Status: **COMPLETE**

Build: 66 routes, 0 TypeScript errors

Production Ready: YES

Documentation: Complete