

# Super Admin Dashboard Guide

## For the Replit Development Agent

Version: 1.0  
Date: 2025-07-24  
Author: Replit Coach Too  
Status: Draft

### 1. Objective

This document provides a detailed technical guide for the Replit Agent to build a **Super Admin Dashboard**. This secure, internal-facing section of the application will be accessible only to the Avallen Solutions team. For the MVP, it will provide core functionalities for platform management, analytics, and quality control.

### 2. Part 1: Backend Implementation (RBAC & Admin APIs)

#### 2.1. Database Enhancement: Role-Based Access Control (RBAC)

To secure the admin section, we must introduce a role system.

- **Modification to users Table:** Add a new column to define user roles.  
| Column Name | Data Type | Constraints | Description |  
| :--- | :--- | :--- | :--- |  
| ... (existing columns) | | | |  
| role | VARCHAR(50) | NOT NULL, DEFAULT 'user' | User's role ('user' or 'admin'). |
- **Admin User Seeding:** A mechanism (e.g., a secure, one-time script) must be created to assign the 'admin' role to the initial Avallen Solutions team members.

#### 2.2. New Admin API Endpoints

- **Requirement:** A new set of API endpoints must be created under the /api/admin/ route.
- **Security:** All endpoints under this route must be protected by a middleware that checks if the authenticated user's role is 'admin'. If not, it must return a 403 Forbidden error.
- **Analytics Endpoint: GET /api/admin/analytics**
  - **Logic:** This endpoint will query the database to compute and return key platform metrics.
  - **Response Body:**

```
{  
  "newUserCount": 150,  
  "userGrowthPercentage": 15.2,  
  "newSupplierCount": 300,
```

```
"supplierGrowthPercentage": 20.0,  
"pendingLcaReviews": 12  
}
```

- **Supplier Management Endpoints:**

- GET /api/admin/suppliers: Fetches a list of ALL suppliers from the verified\_suppliers table, including their verification\_status and who submitted\_by.
- PUT /api/admin/suppliers/<supplier\_id>/verify: Updates a supplier's verification\_status to 'verified'.

- **LCA Approval Endpoints:**

- GET /api/admin/reports/pending: Fetches all reports with status: 'pending\_review'.
- PUT /api/admin/reports/<report\_id>/approve: Updates a report's status to 'approved'.

### 3. Part 2: Frontend Implementation (Admin UI)

A new section of the React application will be created, accessible via a /admin route. This route will be protected and only accessible to users with the 'admin' role.

#### 3.1. Admin Navigation

- **Logic:** In the main SidebarNav component, a new link to "/admin" will be rendered *conditionally*, only if the logged-in user has the 'admin' role.

#### 3.2. Admin Dashboard View (/admin/dashboard)

- **Headline:** "Platform Analytics"
- **UI Components:**
  - A series of AnalyticsCard components, each displaying a key metric from the analytics endpoint (New Users, User Growth, etc.).
  - A QuickActions section with buttons linking to "Verify Suppliers" and "Approve LCAs".

#### 3.3. Supplier Verification View (/admin/suppliers)

- **Headline:** "Supplier Management"
- **UI Component:** AdminSupplierTable
- **Content:** A table displaying all suppliers from the GET /api/admin/suppliers endpoint.
- **Features:**
  - Columns for supplier\_name, submitted\_by, and verification\_status.
  - A filter to show only suppliers with status: 'pending\_review'.

- For each pending supplier, a "Review" button that opens a modal showing the supplier's full details.
- Inside the modal, an **"Approve Supplier"** button that calls the PUT `/api/admin/suppliers/<supplier_id>/verify` endpoint.

### 3.4. LCA Approval Queue (/admin/lca-approvals)

- **Headline:** "LCA Report Approval Queue"
- **UI Component:** ApprovalQueueTable
- **Content:** A table listing all reports from the GET `/api/admin/reports/pending` endpoint.
- **Features:**
  - Columns for `product_name`, `company_name`, and `date_submitted`.
  - A "Review Report" button for each item.
  - Clicking the button navigates to a detailed view where the admin can see the full report data.
  - On the review page, an **"Approve Report"** button that calls the PUT `/api/admin/reports/<report_id>/approve` endpoint.