

SET statement_timeout = 0;

SET lock_timeout = 0;

SET idle_in_transaction_session_timeout = 0;

SET client_encoding = 'UTF8';

SET standard_conforming_strings = on;

SELECT pg_catalog.set_config('search_path', '', false);

SET check_function_bodies = false;

SET xmloption = content;

SET client_min_messages = warning;

SET row_security = off;

CREATE EXTENSION IF NOT EXISTS "pgsodium" WITH SCHEMA "pgsodium";

COMMENT ON SCHEMA "public" IS 'standard public schema';

CREATE EXTENSION IF NOT EXISTS "pg_graphql" WITH SCHEMA "graphql";

CREATE EXTENSION IF NOT EXISTS "pg_stat_statements" WITH SCHEMA "extensions";

CREATE EXTENSION IF NOT EXISTS "pgcrypto" WITH SCHEMA "extensions";

CREATE EXTENSION IF NOT EXISTS "pgjwt" WITH SCHEMA "extensions";

CREATE EXTENSION IF NOT EXISTS "supabase_vault" WITH SCHEMA "vault";

```
CREATE EXTENSION IF NOT EXISTS "uuid-oss" WITH SCHEMA "extensions";
```

```
CREATE TYPE "public"."credit_plan" AS ENUM (  
    'default',  
    'invoice'  
);
```

```
ALTER TYPE "public"."credit_plan" OWNER TO "postgres";
```

```
CREATE TYPE "public"."model_type" AS ENUM (  
    'text',  
    'vision'  
);
```

```
ALTER TYPE "public"."model_type" OWNER TO "postgres";
```

```
CREATE TYPE "public"."organization_member_invite_status" AS ENUM (  
    'waiting',  
    'joined',  
    'expired',  
    'canceled'  
);
```

```
ALTER TYPE "public"."organization_member_invite_status" OWNER TO "postgres";
```

```
CREATE TYPE "public"."organization_member_role" AS ENUM (  
    'waiting',  
    'joined',  
    'expired',  
    'canceled'
```

```
'manager',  
  
'reader'  
);
```

```
ALTER TYPE "public"."organization_member_role" OWNER TO "postgres";
```

```
CREATE TYPE "public"."pricing_plan_interval" AS ENUM (  
  
    'day',  
  
    'week',  
  
    'month',  
  
    'year'  
);
```

```
ALTER TYPE "public"."pricing_plan_interval" OWNER TO "postgres";
```

```
CREATE TYPE "public"."pricing_type" AS ENUM (  
  
    'one_time',  
  
    'recurring'  
);
```

```
ALTER TYPE "public"."pricing_type" OWNER TO "postgres";
```

```
CREATE TYPE "public"."subscription_status" AS ENUM (  
  
    'trialing',  
  
    'active',  
  
    'canceled',
```

```
'incomplete',  
  
'incomplete_expired',  
  
'past_due',  
  
'unpaid',  
  
'paused'  
);
```

```
ALTER TYPE "public"."subscription_status" OWNER TO "postgres";
```

```
CREATE TYPE "public"."user_prompts_status" AS ENUM (  
  
    'approved',  
  
    'pending',  
  
    'rejected'  
);
```

```
ALTER TYPE "public"."user_prompts_status" OWNER TO "postgres";
```

```
CREATE TYPE "public"."user_agents_status" AS ENUM (  
  
    'approved',  
  
    'pending',  
  
    'rejected'  
);
```

```
ALTER TYPE "public"."user_agents_status" OWNER TO "postgres";
```

```
CREATE TYPE "public"."user_swarms_status" AS ENUM (  
  

```

```
'approved',  
'pending',  
'rejected'  
);
```

```
ALTER TYPE "public"."user_swarms_status" OWNER TO "postgres";
```

```
CREATE TYPE "public"."user_tier" AS ENUM (  
    'tier1',  
    'tier2',  
    'tier3',  
    'tier4'  
);
```

```
ALTER TYPE "public"."user_tier" OWNER TO "postgres";
```

```
CREATE TYPE "public"."users_wallets_transaction_type" AS ENUM (  
    'reduct',  
    'add'  
);
```

```
ALTER TYPE "public"."users_wallets_transaction_type" OWNER TO "postgres";
```

```
CREATE OR REPLACE FUNCTION "public"."get_user_id_by_email"("email" "text") RETURNS  
TABLE("id" "uuid")
```

```
LANGUAGE "plpgsql" SECURITY DEFINER
```

```

AS $_$

BEGIN

    RETURN QUERY SELECT au.id FROM auth.users au WHERE au.email = $1;

END;

$_$;


ALTER FUNCTION "public"."get_user_id_by_email"("email" "text") OWNER TO "postgres";


CREATE OR REPLACE FUNCTION "public"."handle_new_user"() RETURNS "trigger"

    LANGUAGE "plpgsql" SECURITY DEFINER

    AS $$begin

insert into public.users (id, full_name, avatar_url, email)

    values (new.id, new.raw_user_meta_data->>'full_name', new.raw_user_meta_data->>'avatar_url',

new.email);

    return new;

end;$$;


ALTER FUNCTION "public"."handle_new_user"() OWNER TO "postgres";


SET default_tablespace = '';


SET default_table_access_method = "heap";


CREATE TABLE IF NOT EXISTS "public"."customers" (

    "id" "uuid" NOT NULL,

    "stripe_customer_id" "text"

```

```
);
```

```
ALTER TABLE "public"."customers" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."invoices" (
```

```
    "id" "text" NOT NULL,
```

```
    "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,
```

```
    "updated_at" timestamp with time zone DEFAULT "now"(),
```

```
    "created" timestamp with time zone,
```

```
    "user_id" "uuid",
```

```
    "stripe_customer_id" "text",
```

```
    "status" "text",
```

```
    "total" bigint,
```

```
    "period_at" timestamp with time zone,
```

```
    "period_end" timestamp with time zone,
```

```
    "metadata" "json",
```

```
    "status_transitions" "json",
```

```
    "is_paid" boolean DEFAULT false,
```

```
    "reason" "text"
```

```
);
```

```
ALTER TABLE "public"."invoices" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."prices" (
```

```
    "id" "text" NOT NULL,
```

```
    "product_id" "text",
```

```
"active" boolean,  
  
"description" "text",  
  
"unit_amount" bigint,  
  
"currency" "text",  
  
"type" "public"."pricing_type",  
  
"interval" "public"."pricing_plan_interval",  
  
"interval_count" integer,  
  
"trial_period_days" integer,  
  
"metadata" "jsonb",  
  
CONSTRAINT "prices_currency_check" CHECK (("char_length"("currency") = 3))  
  
);
```

```
ALTER TABLE "public"."prices" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."products" (  
  
  "id" "text" NOT NULL,  
  
  "active" boolean,  
  
  "name" "text",  
  
  "description" "text",  
  
  "image" "text",  
  
  "metadata" "jsonb"  
  
);
```

```
ALTER TABLE "public"."products" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."subscriptions" (  
  

```



```
"id" "text" NOT NULL,  
  
"user_id" "uuid" NOT NULL,  
  
"status" "public"."subscription_status",  
  
"metadata" "jsonb",  
  
"price_id" "text",  
  
"quantity" integer,  
  
"cancel_at_period_end" boolean,  
  
"created" timestamp with time zone DEFAULT "timezone"('utc'::"text", "now"()) NOT NULL,  
  
"current_period_start" timestamp with time zone DEFAULT "timezone"('utc'::"text", "now"()) NOT  
NULL,  
  
"current_period_end" timestamp with time zone DEFAULT "timezone"('utc'::"text", "now"()) NOT  
NULL,  
  
"ended_at" timestamp with time zone DEFAULT "timezone"('utc'::"text", "now"()),  
  
"cancel_at" timestamp with time zone DEFAULT "timezone"('utc'::"text", "now"()),  
  
"canceled_at" timestamp with time zone DEFAULT "timezone"('utc'::"text", "now"()),  
  
"trial_start" timestamp with time zone DEFAULT "timezone"('utc'::"text", "now"()),  
  
"trial_end" timestamp with time zone DEFAULT "timezone"('utc'::"text", "now"())  
);
```

```
ALTER TABLE "public"."subscriptions" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarm_cloud_billing_transcations" (  
  
"created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
  
"user_id" "uuid",  
  
"total_monthly_cost" double precision,  
  
"stripe_customer_id" "text",
```

```
"transaction_id" "uuid" DEFAULT "gen_random_uuid()",  
"invoice_id" "text",  
"id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,  
"payment_successful" boolean DEFAULT false NOT NULL  
);  
  
ALTER TABLE "public"."swarm_cloud_billing_transcations" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_api_activities" (  
    "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
    "user_id" "uuid" NOT NULL,  
    "api_key_id" "uuid",  
    "model_id" "uuid",  
    "input_tokens" bigint,  
    "output_tokens" bigint,  
    "total_cost" double precision,  
    "input_cost" double precision,  
    "output_cost" double precision,  
    "messages" "json",  
    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,  
    "temperature" real,  
    "top_p" real,  
    "echo" boolean,  
    "stream" boolean,  
    "repetition_penalty" real,  
    "max_tokens" integer,
```

```
"organization_id" "uuid",

"invoice_total_cost" double precision DEFAULT '0'::double precision NOT NULL,

"request_count" bigint DEFAULT '1'::bigint NOT NULL

);

ALTER TABLE "public"."swarms_cloud_api_activities" OWNER TO "postgres";

COMMENT ON COLUMN "public"."swarms_cloud_api_activities"."invoice_total_cost" IS 'Get total
cost of user on the invoice credit_plan';

CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_api_keys" (

    "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,

    "name" "text",

    "user_id" "uuid" NOT NULL,

    "key" "text" NOT NULL,

    "limit_credit_dollar" double precision,

    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,

    "is_deleted" boolean DEFAULT false

);
```

```
ALTER TABLE "public"."swarms_cloud_api_keys" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_models" (

    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,

    "unique_name" "text" NOT NULL,

    "name" "text",
```

```

"description" "text",

"created_at" timestamp with time zone DEFAULT "now"() NOT NULL,

"updated_at" timestamp without time zone,

"creator" "text",

"default_config" "json",

"enabled" boolean DEFAULT true,

"per_1k_input_price" double precision,

"per_1k_output_price" double precision,

"context_length" bigint,

"provider" "text",

"model_parameter" "text",

"docs" "json",

"model_type" "public"."model_type",

"support_functions" boolean DEFAULT false,

"api_endpoint" "text",

"use_cases" "json" DEFAULT '[]'::"json",

"tags" "text",

"model_card_md" "text",

"slug" "text",

"price_million_input" double precision,

"price_million_output" double precision

);

ALTER TABLE "public"."swarms_cloud_models" OWNER TO "postgres";

CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_monthly_usage" (

```

```
"created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
"user_id" "uuid",  
"usage" double precision,  
"api_requests_count" bigint,  
"month" "date",  
"id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL  
);  
  
ALTER TABLE "public"."swarms_cloud_monthly_usage" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_organization_member_invites" (  
    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,  
    "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
    "organization_id" "uuid",  
    "invite_by_user_id" "uuid",  
    "email" "text",  
    "secret_code" "uuid",  
    "status" "public"."organization_member_invite_status",  
    "user_id" "uuid",  
    "role" "public"."organization_member_role"  
);
```

```
ALTER TABLE "public"."swarms_cloud_organization_member_invites" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_organization_members" (  
    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,
```

```

"created_at" timestamp with time zone DEFAULT "now"() NOT NULL,
"organization_id" "uuid",
"user_id" "uuid",
"invite_by_user_id" "uuid",
"is_deleted" boolean,
"deleted_by_user_id" "uuid",
"role" "public"."organization_member_role"
);

ALTER TABLE "public"."swarms_cloud_organization_members" OWNER TO "postgres";

```

```

CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_organizations" (
    "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,
    "owner_user_id" "uuid",
    "name" "text",
    "public_id" "uuid" DEFAULT "gen_random_uuid"(),
    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL
);

```

```

ALTER TABLE "public"."swarms_cloud_organizations" OWNER TO "postgres";

```

```

CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_prompts" (
    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,
    "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,
    "user_id" "uuid" DEFAULT "auth"."uid"() NOT NULL,
    "prompt" "text",

```

```
"name" "text",  
  
"use_cases" "json",  
  
"status" "public"."user_prompts_status",  
  
"tags" "text",  
  
"description" "text"  
);
```

```
ALTER TABLE "public"."swarms_cloud_prompts" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_agents" (  
  
    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,  
  
    "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
  
    "user_id" "uuid" DEFAULT "auth"."uid"() NOT NULL,  
  
    "agent" "text",  
  
    "name" "text",  
  
    "use_cases" "json",  
  
    "agent" "text",  
  
    "status" "public"."user_agents_status",  
  
    "tags" "text",  
  
    "description" "text"  
);
```

```
ALTER TABLE "public"."swarms_cloud_agents" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_rate_limits" (  
  
    "user_id" "uuid" NOT NULL,
```

```
"created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
"request_count" bigint DEFAULT '0'::bigint,  
"last_request_at" timestamp without time zone,  
"id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL  
);  
  
ALTER TABLE "public"."swarms_cloud_rate_limits" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_user_swarms" (  
    "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
    "user_id" "uuid",  
    "pr_id" "text",  
    "pr_link" "text",  
    "code" "text",  
    "name" "text",  
    "description" "text",  
    "tags" "text",  
    "use_cases" "json",  
    "status" "public"."user_swarms_status",  
    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL  
);
```

```
ALTER TABLE "public"."swarms_cloud_user_swarms" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_users_credits" (  
    "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,
```



```
"user_id" "uuid",  
  
"created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
  
"credit" numeric DEFAULT '0'::numeric NOT NULL,  
  
"free_credit" numeric DEFAULT '0'::numeric NOT NULL,  
  
"free_credit_expire_date" timestamp without time zone,  
  
"credit_count" bigint DEFAULT '0'::bigint NOT NULL  
  
);
```

```
ALTER TABLE "public"."swarms_cloud_users_credits" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_users_tiers" (  
  
"id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,  
  
"created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
  
"updated_at" timestamp with time zone,  
  
"user_id" "uuid",  
  
"tier" "public"."user_tier"  
  
);
```

```
ALTER TABLE "public"."swarms_cloud_users_tiers" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_users_wallets" (  
  
"id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,  
  
"created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
  
"balance" bigint DEFAULT '0'::bigint,  
  
"user_id" "uuid" DEFAULT "gen_random_uuid"(),  
  
"name" "text",
```

```
"expire_date" timestamp with time zone,  
"is_deleted" boolean,  
"is_expired" boolean,  
"order" integer  
);
```

```
ALTER TABLE "public"."swarms_cloud_users_wallets" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."swarms_cloud_users_wallets_transactions" (  
  "id" "uuid" DEFAULT "gen_random_uuid"() NOT NULL,  
  "created_at" timestamp with time zone DEFAULT "now"() NOT NULL,  
  "wallet_id" "uuid",  
  "user_id" "uuid",  
  "old_balance" bigint,  
  "new_balance" bigint,  
  "type" "public"."users_wallets_transaction_type",  
  "amount" bigint,  
  "description" "text"  
);
```

```
ALTER TABLE "public"."swarms_cloud_users_wallets_transactions" OWNER TO "postgres";
```

```
CREATE TABLE IF NOT EXISTS "public"."users" (  
  "id" "uuid" NOT NULL,  
  "full_name" "text",  
  "avatar_url" "text",
```

```
"billing_address" "jsonb",  
"payment_method" "jsonb",  
"company_name" "text",  
"job_title" "text",  
"country_code" "text",  
"basic_onboarding_completed" boolean DEFAULT false,  
"signup_reason" "text",  
"referral" "text",  
"about_company" "text",  
"email" "text",  
"credit_fraction" numeric DEFAULT '0'::numeric,  
"credit_plan" "public"."credit_plan" DEFAULT 'default'::"public"."credit_plan" NOT NULL,  
"had_free_credits" boolean DEFAULT false NOT NULL  
);
```

```
ALTER TABLE "public"."users" OWNER TO "postgres";
```

```
COMMENT ON COLUMN "public"."users"."credit_plan" IS 'Set a credit plan for your to choose  
between charging manually or sending invoices at the end of the month.';
```

```
COMMENT ON COLUMN "public"."users"."had_free_credits" IS 'Has user ever received free  
credits?';
```

```
ALTER TABLE ONLY "public"."customers"
```

```
ADD CONSTRAINT "customers_pkey" PRIMARY KEY ("id");
```

ALTER TABLE ONLY "public"."invoices"

ADD CONSTRAINT "invoices_pkey" PRIMARY KEY ("id");

ALTER TABLE ONLY "public"."prices"

ADD CONSTRAINT "prices_pkey" PRIMARY KEY ("id");

ALTER TABLE ONLY "public"."products"

ADD CONSTRAINT "products_pkey" PRIMARY KEY ("id");

ALTER TABLE ONLY "public"."subscriptions"

ADD CONSTRAINT "subscriptions_pkey" PRIMARY KEY ("id");

ALTER TABLE ONLY "public"."swarm_cloud_billing_transcations"

ADD CONSTRAINT "swarm_cloud_billing_transcations_id_key" UNIQUE ("id");

ALTER TABLE ONLY "public"."swarm_cloud_billing_transcations"

ADD CONSTRAINT "swarm_cloud_billing_transcations_pkey" PRIMARY KEY ("id");

ALTER TABLE ONLY "public"."swarms_cloud_api_activities"

ADD CONSTRAINT "swarms_cloud_api_activities_pkey" PRIMARY KEY ("id");

ALTER TABLE ONLY "public"."swarms_cloud_api_keys"

ADD CONSTRAINT "swarms_cloud_api_keys_key_key" UNIQUE ("key");

ALTER TABLE ONLY "public"."swarms_cloud_api_keys"

ADD CONSTRAINT "swarms_cloud_api_keys_pkey" PRIMARY KEY ("id");

```
ALTER TABLE ONLY "public"."swarms_cloud_models"
```

```
ADD CONSTRAINT "swarms_cloud_models_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_monthly_usage"
```

```
ADD CONSTRAINT "swarms_cloud_monthly_usage_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_member_invites"
```

```
ADD CONSTRAINT "swarms_cloud_organization_member_invites_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_members"
```

```
ADD CONSTRAINT "swarms_cloud_organization_members_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organizations"
```

```
ADD CONSTRAINT "swarms_cloud_organizations_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_prompts"
```

```
ADD CONSTRAINT "swarms_cloud_prompts_id_key" UNIQUE ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_prompts"
```

```
ADD CONSTRAINT "swarms_cloud_prompts_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_agents"
```

```
ADD CONSTRAINT "swarms_cloud_agents_id_key" UNIQUE ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_agents"
```

```
ADD CONSTRAINT "swarms_cloud_agents_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_rate_limits"
```

```
ADD CONSTRAINT "swarms_cloud_rate_limits_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_user_swarms"
```

```
ADD CONSTRAINT "swarms_cloud_user_swarms_name_key" UNIQUE ("name");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_user_swarms"
```

```
ADD CONSTRAINT "swarms_cloud_user_swarms_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_users_credits"
```

```
ADD CONSTRAINT "swarms_cloud_users_credits_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_users_credits"
```

```
ADD CONSTRAINT "swarms_cloud_users_credits_user_id_key" UNIQUE ("user_id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_users_tiers"
```

```
ADD CONSTRAINT "swarms_cloud_users_tiers_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_users_wallets"
```

```
ADD CONSTRAINT "swarms_cloud_users_wallets_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_users_wallets_transactions"
```

```
ADD CONSTRAINT "swarms_cloud_users_wallets_transactions_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."users"
```

```
ADD CONSTRAINT "users_pkey" PRIMARY KEY ("id");
```

```
ALTER TABLE ONLY "public"."swarm_cloud_billing_transcations"
```

```
ADD CONSTRAINT "Swarm Cloud Billing Transcations_user_id_fkey" FOREIGN KEY ("user_id")  
REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."prices"
```

```
ADD CONSTRAINT "prices_product_id_fkey" FOREIGN KEY ("product_id") REFERENCES  
"public"."products"("id");
```

```
ALTER TABLE ONLY "public"."customers"
```

```
ADD CONSTRAINT "public_customers_id_fkey" FOREIGN KEY ("id") REFERENCES  
"auth"."users"("id") ON DELETE CASCADE;
```

```
ALTER TABLE ONLY "public"."invoices"
```

```
ADD CONSTRAINT "public_invoices_user_id_fkey" FOREIGN KEY ("user_id") REFERENCES  
"public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_api_activities"
```

```
ADD CONSTRAINT "public_swarms_cloud_api_activities_api_key_id_fkey" FOREIGN KEY  
("api_key_id") REFERENCES "public"."swarms_cloud_api_keys"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_api_activities"
```

```
ADD CONSTRAINT "public_swarms_cloud_api_activities_model_id_fkey" FOREIGN KEY  
("model_id") REFERENCES "public"."swarms_cloud_models"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_api_activities"
```

```
    ADD CONSTRAINT "public_swarms_cloud_api_activities_organization_id_fkey" FOREIGN KEY  
("organization_id") REFERENCES "public"."swarms_cloud_organizations"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_api_activities"
```

```
    ADD CONSTRAINT "public_swarms_cloud_api_activities_user_id_fkey" FOREIGN KEY  
("user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_api_keys"
```

```
    ADD CONSTRAINT "public_swarms_cloud_api_keys_user_id_fkey" FOREIGN KEY ("user_id")  
REFERENCES "public"."users"("id") ON DELETE CASCADE;
```

```
ALTER TABLE ONLY "public"."swarms_cloud_monthly_usage"
```

```
    ADD CONSTRAINT "public_swarms_cloud_monthly_usage_user_id_fkey" FOREIGN KEY  
("user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_member_invites"
```

```
    ADD CONSTRAINT "public_swarms_cloud_organization_member_invites_invite_by_user_"  
FOREIGN KEY ("invite_by_user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_member_invites"
```

```
    ADD CONSTRAINT "public_swarms_cloud_organization_member_invites_organization_id"  
FOREIGN KEY ("organization_id") REFERENCES "public"."swarms_cloud_organizations"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_member_invites"
```



```
ADD CONSTRAINT "public_swarms_cloud_organization_member_invites_user_id_fkey"  
FOREIGN KEY ("user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_members"
```

```
ADD CONSTRAINT "public_swarms_cloud_organization_members_deleted_by_user_id_fke"  
FOREIGN KEY ("deleted_by_user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_members"
```

```
ADD CONSTRAINT "public_swarms_cloud_organization_members_invited_by_user_id_fke"  
FOREIGN KEY ("invite_by_user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_members"
```

```
ADD CONSTRAINT "public_swarms_cloud_organization_members_organization_id_fkey"  
FOREIGN KEY ("organization_id") REFERENCES "public"."swarms_cloud_organizations"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organization_members"
```

```
ADD CONSTRAINT "public_swarms_cloud_organization_members_user_id_fkey" FOREIGN  
KEY ("user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_organizations"
```

```
ADD CONSTRAINT "public_swarms_cloud_organizations_owner_user_id_fkey" FOREIGN KEY  
("owner_user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_user_swarms"
```

```
ADD CONSTRAINT "public_swarms_cloud_user_swarms_user_id_fkey" FOREIGN KEY  
("user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_users_credits"  
  
    ADD CONSTRAINT "public_swarms_cloud_users_credits_user_id_fkey" FOREIGN KEY  
("user_id") REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_users_tiers"  
  
    ADD CONSTRAINT "public_swarms_cloud_users_tiers_user_id_fkey" FOREIGN KEY ("user_id")  
REFERENCES "public"."users"("id") ON DELETE CASCADE;
```

```
ALTER TABLE ONLY "public"."subscriptions"  
  
    ADD CONSTRAINT "subscriptions_price_id_fkey" FOREIGN KEY ("price_id") REFERENCES  
"public"."prices"("id");
```

```
ALTER TABLE ONLY "public"."subscriptions"  
  
    ADD CONSTRAINT "subscriptions_user_id_fkey" FOREIGN KEY ("user_id") REFERENCES  
"auth"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_prompts"  
  
    ADD CONSTRAINT "swarms_cloud_prompts_user_id_fkey" FOREIGN KEY ("user_id")  
REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_agents"  
  
    ADD CONSTRAINT "swarms_cloud_agents_user_id_fkey" FOREIGN KEY ("user_id")  
REFERENCES "public"."users"("id");
```

```
ALTER TABLE ONLY "public"."swarms_cloud_rate_limits"
```

```
ADD CONSTRAINT "swarms_cloud_rate_limits_user_id_fkey" FOREIGN KEY ("user_id")
REFERENCES "public"."users"("id") ON UPDATE CASCADE ON DELETE CASCADE;

ALTER TABLE ONLY "public"."swarms_cloud_users_wallets_transactions"

ADD CONSTRAINT "swarms_cloud_users_wallets_transactions_user_id_fkey" FOREIGN KEY
("user_id") REFERENCES "public"."users"("id");

ALTER TABLE ONLY "public"."swarms_cloud_users_wallets_transactions"

ADD CONSTRAINT "swarms_cloud_users_wallets_transactions_wallet_id_fkey" FOREIGN KEY
("wallet_id") REFERENCES "public"."swarms_cloud_users_wallets"("id");

ALTER TABLE ONLY "public"."swarms_cloud_users_wallets"

ADD CONSTRAINT "swarms_cloud_users_wallets_user_id_fkey" FOREIGN KEY ("user_id")
REFERENCES "public"."users"("id");

ALTER TABLE ONLY "public"."users"

ADD CONSTRAINT "users_id_fkey" FOREIGN KEY ("id") REFERENCES "auth"."users"("id");

CREATE POLICY "Allow public read-only access." ON "public"."prices" FOR SELECT USING (true);

CREATE POLICY "Allow public read-only access." ON "public"."products" FOR SELECT USING
(true);

CREATE POLICY "Can only view own subs data." ON "public"."subscriptions" FOR SELECT
USING (("auth"."uid"() = "user_id"));
```

```
CREATE POLICY "Can update own user data." ON "public"."users" FOR UPDATE USING  
(("auth"."uid"() = "id"));
```

```
CREATE POLICY "Can view own user data." ON "public"."users" FOR SELECT USING  
(("auth"."uid"() = "id"));
```

```
ALTER TABLE "public"."customers" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."invoices" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."prices" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."products" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."subscriptions" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarm_cloud_billing_transcations" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_api_activities" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_api_keys" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_models" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_monthly_usage" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_organization_member_invites" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_organization_members" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_organizations" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_prompts" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_agents" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_rate_limits" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_user_swarms" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_users_credits" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_users_tiers" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_users_wallets" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."swarms_cloud_users_wallets_transactions" ENABLE ROW LEVEL SECURITY;
```

```
ALTER TABLE "public"."users" ENABLE ROW LEVEL SECURITY;
```

ALTER PUBLICATION "supabase_realtime" OWNER TO "postgres";

ALTER PUBLICATION "supabase_realtime" ADD TABLE ONLY "public"."prices";

ALTER PUBLICATION "supabase_realtime" ADD TABLE ONLY "public"."products";

GRANT USAGE ON SCHEMA "public" TO "postgres";

GRANT USAGE ON SCHEMA "public" TO "anon";

GRANT USAGE ON SCHEMA "public" TO "authenticated";

GRANT USAGE ON SCHEMA "public" TO "service_role";

GRANT ALL ON FUNCTION "public"."get_user_id_by_email"("email" "text") TO "anon";

GRANT ALL ON FUNCTION "public"."get_user_id_by_email"("email" "text") TO "authenticated";

GRANT ALL ON FUNCTION "public"."get_user_id_by_email"("email" "text") TO "service_role";

GRANT ALL ON FUNCTION "public"."handle_new_user"() TO "anon";

GRANT ALL ON FUNCTION "public"."handle_new_user"() TO "authenticated";

GRANT ALL ON FUNCTION "public"."handle_new_user"() TO "service_role";

GRANT ALL ON TABLE "public"."customers" TO "anon";

GRANT ALL ON TABLE "public"."customers" TO "authenticated";

GRANT ALL ON TABLE "public"."customers" TO "service_role";

GRANT ALL ON TABLE "public"."invoices" TO "anon";

GRANT ALL ON TABLE "public"."invoices" TO "authenticated";

GRANT ALL ON TABLE "public"."invoices" TO "service_role";

GRANT ALL ON TABLE "public"."prices" TO "anon";

GRANT ALL ON TABLE "public"."prices" TO "authenticated";

GRANT ALL ON TABLE "public"."prices" TO "service_role";

GRANT ALL ON TABLE "public"."products" TO "anon";

GRANT ALL ON TABLE "public"."products" TO "authenticated";

GRANT ALL ON TABLE "public"."products" TO "service_role";

GRANT ALL ON TABLE "public"."subscriptions" TO "anon";

GRANT ALL ON TABLE "public"."subscriptions" TO "authenticated";

GRANT ALL ON TABLE "public"."subscriptions" TO "service_role";

GRANT ALL ON TABLE "public"."swarm_cloud_billing_transcations" TO "anon";

GRANT ALL ON TABLE "public"."swarm_cloud_billing_transcations" TO "authenticated";

GRANT ALL ON TABLE "public"."swarm_cloud_billing_transcations" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_api_activities" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_api_activities" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_api_activities" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_api_keys" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_api_keys" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_api_keys" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_models" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_models" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_models" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_monthly_usage" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_monthly_usage" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_monthly_usage" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_organization_member_invites" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_organization_member_invites" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_organization_member_invites" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_organization_members" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_organization_members" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_organization_members" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_organizations" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_organizations" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_organizations" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_prompts" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_prompts" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_prompts" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_agents" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_agents" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_agents" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_rate_limits" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_rate_limits" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_rate_limits" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_user_swarms" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_user_swarms" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_user_swarms" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_users_credits" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_users_credits" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_users_credits" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_users_tiers" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_users_tiers" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_users_tiers" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_users_wallets" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_users_wallets" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_users_wallets" TO "service_role";

GRANT ALL ON TABLE "public"."swarms_cloud_users_wallets_transactions" TO "anon";

GRANT ALL ON TABLE "public"."swarms_cloud_users_wallets_transactions" TO "authenticated";

GRANT ALL ON TABLE "public"."swarms_cloud_users_wallets_transactions" TO "service_role";

GRANT ALL ON TABLE "public"."users" TO "anon";

GRANT ALL ON TABLE "public"."users" TO "authenticated";

GRANT ALL ON TABLE "public"."users" TO "service_role";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON SEQUENCES TO "postgres";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON SEQUENCES TO "anon";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON SEQUENCES TO "authenticated";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON SEQUENCES TO "service_role";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON FUNCTIONS TO "postgres";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON FUNCTIONS TO "anon";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON FUNCTIONS TO "authenticated";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON FUNCTIONS TO "service_role";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON TABLES TO "postgres";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON TABLES TO "anon";

ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON

```
TABLES TO "authenticated";
```

```
ALTER DEFAULT PRIVILEGES FOR ROLE "postgres" IN SCHEMA "public" GRANT ALL ON  
TABLES TO "service_role";
```

```
RESET ALL;
```