**SISTEM FIDS – flight information display system**

**Tabele Postgres**

Server address: 192.168.0.126

UserName: postgres

UserPassword: Alramlah1978

Database Name: FIDS

User-ul “postgres” pentru conectare la server si la baza de date

Client admin Postgres: <https://www.pgadmin.org/download/>

Tabele:

CREATE TABLE public."CheckIN"

(

"CheckIN\_ID" integer NOT NULL DEFAULT nextval('"CheckIN\_CheckIN\_ID\_seq"'::regclass),

"Display\_ID" integer,

"Flight\_ID" integer,

"Status" integer,

"Is\_Deleted" boolean,

CONSTRAINT "CheckIN\_pkey" PRIMARY KEY ("CheckIN\_ID")

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

CREATE TABLE public."Companies"

(

"Company\_ID" integer NOT NULL DEFAULT nextval('"Companies\_Company\_ID\_seq"'::regclass),

"Company\_Name" text COLLATE pg\_catalog."default" NOT NULL,

"Company\_Code" text COLLATE pg\_catalog."default" NOT NULL,

"Company\_Logo" text COLLATE pg\_catalog."default",

"Is\_Deleted" boolean,

CONSTRAINT "Companies\_pkey" PRIMARY KEY ("Company\_ID")

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

CREATE TABLE public."Destinations"

(

"City\_ID" integer NOT NULL DEFAULT nextval('"Destinations\_City\_ID\_seq"'::regclass),

"City\_Name" text COLLATE pg\_catalog."default" NOT NULL,

"City\_Code" text COLLATE pg\_catalog."default" NOT NULL,

"Is\_Deleted" boolean,

CONSTRAINT "Destinations\_pkey" PRIMARY KEY ("City\_ID")

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

CREATE TABLE public."Displays"

(

"Display\_ID" integer NOT NULL DEFAULT nextval('"Displays\_Display\_ID\_seq"'::regclass),

"Display\_Name" text COLLATE pg\_catalog."default",

"Display\_URL" text COLLATE pg\_catalog."default",

"Display\_Group" text COLLATE pg\_catalog."default",

CONSTRAINT "Displays\_pkey" PRIMARY KEY ("Display\_ID")

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

CREATE TABLE public."Flights"

(

"Flight\_ID" integer NOT NULL DEFAULT nextval('"Flights\_Flight\_ID\_seq"'::regclass),

"Destination\_ID" integer NOT NULL,

"Origin\_ID" integer NOT NULL,

"Company\_ID" integer NOT NULL,

"Flight\_No" text COLLATE pg\_catalog."default" NOT NULL,

"Flight\_Date" date,

"Dest\_Est\_Hour" time without time zone,

"Dest\_Real\_Hour" time without time zone,

"Arrival\_Est\_Hour" time with time zone,

"Arrival\_Real\_Hour" time without time zone,

"Depart\_Delay" text COLLATE pg\_catalog."default",

"Arrival\_Delay" text COLLATE pg\_catalog."default",

"Terminal" text COLLATE pg\_catalog."default",

"Flight\_Type1" integer,

"Flyght\_Type2" integer,

"Cancelled" boolean,

"Status" text COLLATE pg\_catalog."default",

"Is\_Deleted" boolean,

CONSTRAINT "Flights\_pkey" PRIMARY KEY ("Flight\_ID")

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

CREATE TABLE public."Gates"

(

"Gate\_ID" integer NOT NULL DEFAULT nextval('"Gates\_Gate\_ID\_seq"'::regclass),

"Display\_ID" integer,

"Flight\_ID" integer,

"Status" integer,

"Is\_Deleted" boolean,

CONSTRAINT "Gates\_pkey" PRIMARY KEY ("Gate\_ID")

)

WITH (

OIDS = FALSE

)

TABLESPACE pg\_default;

**Functii**.

1. Adaugare/Editare/Stergere zbor

* De adaugat toate informatiile
* Cand se alege intervalul zborului:
  + Interval de timp
  + Check-box cu toate zilele in care se zboara
  + Choice-box cu all-days, week days, weekend-days – corelat automat cu check-box-urile de zile

1. Editare intarziere/anulare zbor

* Cand se schimba ora reala de aterizare/decolare – automat se schimba si statusul in intarziat
* Daca ora reala este egala cu ora estimate, atunci statusul este On Time
* Daca ora curenta este mai mare cu un interval de timp predefinit (de ex. 5 min), atunci statusul se schimba automat in aterizat/decolat
* Posibilitate din interfata de a seta campul Cancelled, in aceasta situatie nu se mai tine cont de ora reala de decolare/aterizare

1. Editare Check-in

* In interfata se afiseaza zborurile din ziua curenta
* Se alege zborul si display-urile pe care sa fie afisat, cu status default Open
* Posibilitate de a schimba statusul din interfata
* De vazut ce afisam cand nu avem informatii

1. Editare poarta de imbarcare

* Se afiseaza zborurile curente
* Se alege zborul pentru fiecare poarta de imbarcare
* De vazut ce afisam cand nu avem informatii

1. Informatii bagaje nerne si international se iau direct din baza de date

* Se afiseaza toate zborurile aterizate (status aterizat), cu un gap de timp de 1.5h – parametru modificabil

1. User

* Predefiniti, doua roluri
* Admin – drepturi totale
* Operator – drepturi:
  + editare ora reala,
  + cancel flight,
  + status Check-in,
  + asociere check-in/display,
  + asociere poarta/display

**Tehnologii folosite**

1. MagicInfo:

* Setare URL pentru fiecare display
* Setare playlist, daca se doreste si afisare pictures/movies
* Afisare alternativa sosiri/plecari

1. Postgress
2. Apache
3. Django
4. Angular

* Pentru refresh automat de pagina in momentul in care se fac schimbari in baza de date