CREATE TABLE User(

email VARCHAR(64) NOT NULL PRIMARY KEY,

password VARCHAR(64) NOT NULL,

nickname VARCHAR(64),

notes VARCHAR(1024)

);

CREATE TABLE Contact(

ID SERIAL NOT NULL PRIMARY KEY,

name VARCHAR(32) NOT NULL,

surname VARCHAR(32) NOT NULL,

propic VARCHAR(1024) DEFAULT ‘C:\Users\Velios\Desktop\Uni\cd\defaultpic’

favorite BOOLEAN NOT NULL DEFAULT ‘false’

user string NOT NULL,

CONSTRAINT FK\_userContact FOREIGN KEY (user) REFERENCES User(email)

);

CREATE TABLE Address(

street VARCHAR(1024) NOT NULL,

zipCode CHAR(5) NOT NULL,

city VARCHAR(32),

province CHAR(2),

country VARCHAR(32),

CONSTRAINT PK\_Address PRIMARY KEY (street, zipcode)

);

CREATE TABLE Email(

email VARCHAR(1024) NOT NULL PRIMARY KEY,

main BOOLEAN NOT NULL

contact

);

CREATE TABLE Social(

supplierName VARCHAR(64) NOT NULL

nickname VARCHAR(64) NOT NULL,

assEmail VARCHAR(1024) NOT NULL,

status VARCHAR(1024),

CONSTRAINT PK\_Social PRIMARY KEY (supplierName, assEmail),

COSTRAINT FK\_AssEmail FOREIGN KEY assEmail REFERENCES Email(email) ON DELETE CASCADE

);

CREATE TABLE PhoneNumber(

type VARCHAR(8) NOT NULL,

number CHAR(10) NOT NULL PRIMARY KEY,

linkedNumber BOOLEAN NOT NULL,

);

CREATE TABLE Group(

groupId SERIAL NOT NULL PRIMARY KEY,

groupName VARCHAR(32) NOT NULL,

creationDate DATE NOT NULL

description VARCHAR(64),

user

);

CREATE TABLE Participants(

groupID VARCHAR(5) NOT NULL ,

contactID VARCHAR(5) NOT NULL ,

joinDate DATE NOT NULL,

CONSTRAINT FK\_GroupID FOREIGN KEY (GroupID) REFERENCES Group(ID) ON DELETE CASCADE

CONSTRAINT FK\_ContactID FOREIGN KEY (contactID) REFERENCES Contact(ID) ON DELETE CASCADE

);

CREATE TABLE AssignedAddress(

contactId SERIAL NOT NULL,

addressStr VARCHAR(126) NOT NULL,

addressZip VARCHAR(5) NOT NULL,

main BOOLEAN DEFAULT ‘false’,

CONSTRAINT FK\_ADContactID FOREING KEY (contactID) REFERENCES Contact(ID) ON DELETE

CASCADE,

CONSTRAINT FK\_ADAddress FOREIGN KEY (addressStr, addressZip) REFERENCES Address(street,

zipCode) ON DELETE CASCADE

)

CREATE TABLE AssignedPhone(

phoneNumber CHAR(10) NOT NULL,

contactId SERIAL NOT NULL,

CONSTRAINT FK\_APContactID FOREIGN KEY (ContactId) REFERENCES Contact(ID) ON DELETE

CASCADE,

CONSTRAINT FK\_phonenumber FOREGIN KEY(phoneNumber) REFERENCES PhoneNumber(number)

ON DELETE CASCADE

)

CREATE TABLE Call(

callID serial NOT NULL PRIMARY KEY,

callType string NOT NULL,

dateH datetime NOT NULL,

number char(10) NOT NULL,

duration INTERVAL (MINUTE, SECOND),

contactName string DEFATULT ‘UNKNOWN’,

user string NOT NULL,

CONSTRAINT FK\_CallContact FOREIGN KEY (user) REFERENCES User(email) ON DELETE CASCADE

)

**VINCOLI**

ALTER TABLE Email

ADD CONSTRAINT controllo\_email CHECK ( email LIKE ‘\_%@\_%.\_\_%’)

ALTER TABLE User (

ADD CONSTRAINT controllo\_pw CHECK (LENGTH(password)>7),

ADD CONSTRAINT controllo\_nick CHECK (LENGTH(nickname)>2)

)

CREATE TRIGGER **uniqueMainAddress,**

BEFOREINSERT OR UPDATE ON AssignedAddress

FOR EACH ROW

BEGIN

DECLARE

temp integer := 0;

IF NEW.main = TRUE

SELECT COUNT(\*) INTO temp

FROM AssignedAddress AS AA

WHERE NEW.contactID = AA.contactID AND main = true

IF (temp > 0) THEN RAISE\_APPLICATION\_ERROR( ?? , ‘Main address for this Contact

already exists!’)

END IF

END