CREATE TABLE UserR(

email VARCHAR(64) NOT NULL PRIMARY KEY,

passW VARCHAR(64) NOT NULL,

nickname VARCHAR(64),

notes VARCHAR(1024)

);

CREATE TABLE Contact(

contactID SERIAL NOT NULL PRIMARY KEY,

contactName 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',

userR VARCHAR(64) NOT NULL,

CONSTRAINT FK\_userContact FOREIGN KEY (UserR) REFERENCES UserR(email) ON DELETE CASCADE

);

CREATE TABLE GroupC(

groupID SERIAL NOT NULL PRIMARY KEY,

groupName VARCHAR(32) NOT NULL,

creationDate DATE NOT NULL

description VARCHAR(64),

userR VARCHAR(64) NOT NULL,

CONSTRAINT FK\_userGroup FOREIGN KEY (userR) REFERENCES UserR(email) ON DELETE CASCADE

);

CREATE TABLE Participant(

groupID VARCHAR(5) NOT NULL,

contactID VARCHAR(5) NOT NULL,

joinDate DATE NOT NULL,

CONSTRAINT FK\_groupIDParticipant FOREIGN KEY (groupID) REFERENCES GroupR(groupID) ON DELETE CASCADE

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

);

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 DEFAULT 'false',

contactID VARCHAR(64),

CONSTRAINT FK\_contactEmail FOREIGN KEY (contactID) REFERENCES Contact(contactID) ON DELETE NULL

);

CREATE TABLE Messaging(

supplierName VARCHAR(64) NOT NULL

nickname VARCHAR(64) NOT NULL,

assEmail VARCHAR(1024) NOT NULL,

status VARCHAR(1024),

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

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

);

CREATE TABLE PhoneNumber(

phoneType VARCHAR(8) NOT NULL,

phoneNumber CHAR(10) NOT NULL PRIMARY KEY,

linkedNumber BOOLEAN NOT NULL DEFAULT 'false',

);

CREATE TABLE PhoneCall(

callID SERIAL NOT NULL PRIMARY KEY,

callType VARCHAR(32) NOT NULL,

dateH TIMESTAMP NOT NULL,

phoneNumber CHAR(10) NOT NULL,

duration INTERVAL,

contactName VARCHAR(64) DEFAULT 'UNKNOWN',

userR VARCHAR(64) NOT NULL,

CONSTRAINT FK\_userCall FOREIGN KEY (userR) REFERENCES UserR(email) ON DELETE CASCADE

);

CREATE TABLE AssignedAddress(

contactID INTEGER NOT NULL,

addressStr VARCHAR(126) NOT NULL,

addressZip VARCHAR(5) NOT NULL,

main BOOLEAN DEFAULT 'false',

CONSTRAINT FK\_ADContactID FOREING KEY (contactID) REFERENCES Contact(contactID) 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 INTEGER NOT NULL,

CONSTRAINT FK\_APContactID FOREIGN KEY (ContactId) REFERENCES Contact(contactID) ON DELETE CASCADE,

CONSTRAINT FK\_phoneNumber FOREIGN KEY(phoneNumber) REFERENCES PhoneNumber(phoneNumber) 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