**Passenger – Login**

userID = getUserName();

pass = getPassword();

userFromDB = PassengerTable.select(userID);

if(hash-md5(pass) == userFromDB.Password){

userFromDB.isLogin = true;

PassengerTable.update(userID,userFromDB);

Return(“login successful”);

}else{

Return(“invalid login”);

}

**Passenger – Logout**

userID = getUserName();

userFromDB = PassengerTable.select(userID);

if(userFromDB.isLogin){

userFromDB.isLogin = false;

PassengerTable.update(userID,userFromDB);

Return(“logout successful”);

}else{

Return(“invalid user”);

}

**Passenger – Register**

userID = getUserName();

pass = getPassword();

name = getName();

u = PassengerTable.select(userID);

if(u == null){

PassengerTable.insert(userID,name,pass,0,false);

Return(“registered successful”);

}else{

Return(“this username is already registered”);

}

**Passenger – DeleteAccount**

userID = getUserName();

pass = getPassword();

userFromDB = PassengerTable.select(userID);

if(userFromDB.isLogin && hash-md5(pass) == userFromDB.Password){

PassengerTable.delete(userFromDB);

Return(“successfully deleted”);

}else{

Return(“unsuccessful”);

}

**Passenger – Choose Origin / Destination**

userID = getUserName();

orig = getOrigin();

dest = getDestination();

userFromDB = PassengerTable.select(userID);

if(userFromDB.isLogin){

Return calculatePrice(orig,dest);

}else{

Return(“unsuccessful”);

}

**Passenger – RequestTravel**

uID = getUserName();

orig = getOrigin();

dest = getDestination();

userFromDB = PassengerTable.select(userID);

if(userFromDB.isLogin){

tID = getNewTravelID();

price = calculatePrice(orig,dest);

TravelTable.insert(tID,null,uID,orig,dest,price);

Return(“travel submitted”,tID);

}else{

Return(“unsuccessful”);

}

**Passenger – CancelTravel**

tID = getTravelID();

travelFromDB = TravelsTable.select(tID);

u=travelFromDB.UserID();

if(u.isLogin){

TravelsTable.delete(travelFromDB);

Return(“travel cancelled !”);

}else{

Return(“unsuccessful”);

}

**Passenger – IncreaseCredits**

uID = getUserName();

amount = getAmount();

bv = getBankVerificationCode();

u = PassengersTable.select(uID);

if(u.isLogin && verifyBankTransaction(bv,amount)){

If(verifyBankTransaction(bv,amount)){

u.Credits += amount;

PassengersTable.update(uID,u);

trID = getNewTransactionID();

TransactionLog.insert(trID,uID,amount,getDateTime(),bv);

Return(“credits added successfully”);

}

Return(“invalid bank transaction”);

}else{

Return(“unsuccessful”);

}

**Passenger – CheckNotification**

uID = getActiveUserName();

u = PassengersTable.select(uID);

if(u.isLogin){

notif = PassengerNotificationTable.select(uID).getOldest();

if(notif.NotificationCode == 1){

DriverFound();

Return(“Driver Found”);

}else if(notif.NotificationCode == 2){

TravelFinnished();

Return(“Travel Finished”);

}

Return(“Unknown Notification”);

}else{

Return(“unsuccessful”);

}

**Driver – Login**

userID = getUserName();

pass = getPassword();

userFromDB = DriverTable.select(userID);

if(hash-md5(pass) == userFromDB.Password){

userFromDB.isLogin = true;

DriverTable.update(userID, userFromDB);

Return(“login successful”);

}else{

Return(“invalid login”);

}

**Driver – Logout**

userID = getUserName();

userFromDB = DriverTable.select(userID);

if(userFromDB.isLogin){

userFromDB.isLogin = false;

DriverTable.update(userID, userFromDB);

Return(“logout successful”);

}else{

Return(“invalid user”);

}

**Driver – Ready For Travel**

userID = getUserName();

userFromDB = DriverTable.select(userID);

if(userFromDB.isLogin){

userFromDB.isReadyForTravel = true;

DriverTable.update(userID, userFromDB);

Return(“successful”);

}else{

Return(“invalid login”);

}

**Driver – Not Ready For Travel**

userID = getUserName();

userFromDB = DriverTable.select(userID);

if(userFromDB.isLogin){

userFromDB.isReadyForTravel = false;

DriverTable.update(userID, userFromDB);

Return(“successful”);

}else{

Return(“invalid login”);

}

**Driver – Check Notification**

userID = getActiveUserName();

userFromDB = DriverTable.select(userID);

if(userFromDB.isLogin){

notif = DriverNotificationTable.select(uID).getOldest();

if(notif.NotificationCode == 1){

PassengerFound();

Return(“Passenger Found”);

}

Return(“Unknown Notification”);

}else{

Return(“invalid login”);

}

**Driver – Travel Finished**

tID = getTravelID();

t = TravelsTable.select(tID);

if(t.DriverID.isLogin){

t.isFinished = true;

TravelsTable.update(tID,t);

}else{

Return(“invalid login”);

}

**Driver – Passenger Found**

uID = getUserName();

userFromDB = DriverTable.select(uID);

if(userFromDB.isLogin){

t = TravelsTable.select(Driver := uID , isFinished := false)[0];

return(t.tID,t.Origin,t.Destination);

Return(“successful”);

}else{

Return(“invalid login”);

}

**Server – Register Driver**

uID = getUserName();

pass = getPassword();

ct = getCarType();

cp = getPlate();

ni = getNationalCode();

DriversTable.insert(uID,pass,ct,cp,ni);

**Server – Assign Driver**

Ut = TravelsTable.select( Driver := null )

for(Travel t : Ut) {

d = getClosestDriver(t.Origin);

TravelsTable.update(t, Driver <= d);

DriverNotificationTable.insert(getNewNotificationID(),t.Driver,1);

PassengerNotificationTable.insert(getNewNotificationID(),t.Passenger,1);

}