



True Pass Database Models

Relational Data Model

Implemented using Firebase
iOS and Android



RELATIONAL DATA MODEL

TPUser

<u>userIdentifier</u>	firstName	lastName	email	imageRef
-----------------------	-----------	----------	-------	----------

TPLocation

<u>locationIdentifier</u>	title	shortTitle	longitude	latitude	locationType	affiliationAccessCode	geofenceRadius	isOpen	imageRef
---------------------------	-------	------------	-----------	----------	--------------	-----------------------	----------------	--------	----------

Pass (weak entity owned by its userIdentifier)

<u>passIdentifier</u>	locationIdentifier	firstName	lastName	email	phoneNumber	startDate	endDate	isActive	didCheckIn	accessCodeQR	imageRef
-----------------------	--------------------	-----------	----------	-------	-------------	-----------	---------	----------	------------	--------------	----------

TPAffiliation

<u>locationIdentifier</u>	<u>userIdentifier</u>	isAdministrator	canGrantPasses	canGrantEntry	canControlUsers	hasAccess	accessCodeQR
---------------------------	-----------------------	-----------------	----------------	---------------	-----------------	-----------	--------------

TPCheckInStamp

<u>userIdentifier</u>	<u>locationIdentifier</u>	timeStampIn	timeStampOut
-----------------------	---------------------------	-------------	--------------



DATABASE DETAILS

Firestore

- Firestore stores all data as a JSON text file, mapping keys to values in a parent-node / child-node format.
- Each relation above uses surrogate keys generated by Firestore as its primary key.
- Each TPUser's 'userIdentifier' is generated by FirebaseAuth when a user creates an account. All 'imageRef' attributes above refer to a link into FirebaseStorage, where images can be located.
- Firestore is flexible and allows new child data to be added at any time, so each of these relation's schema can be modified at any time. Thus, the above data model is not static and can evolve over time as the application does.

Relationships

- TPUsers affiliate with TPLocations using TPAffiliations because each user can affiliate with multiple locations.
 - TPUsers must check-in using TPCheckInStamps for the same reason; their checking in must be associated with a location.
 - 'Pass' is short for TruePass as to avoid confusion with the application name. Passes are weak entities that are owned by the TPUser that granted them. Thus, the complete primary key for each pass is userIdentifier & passIdentifier
-