FamilyCart

A shopping cart mobile application
with real-time data synchronization

Saubhik Mukherjee (I-9)
saubhik@gatech.edu

FamilyCart

Simple ***shared*** shopping list ***for households*** providing

- ***real-time*** data ***synchronization***
- among ***mobile devices***
- with ***offline usability***
- intuitive & modern ***user interface***
- with simple ***spending statistics***

Technologies

Swift

python™

iOS 15

Firebase
Realtime Database

{ JSON }

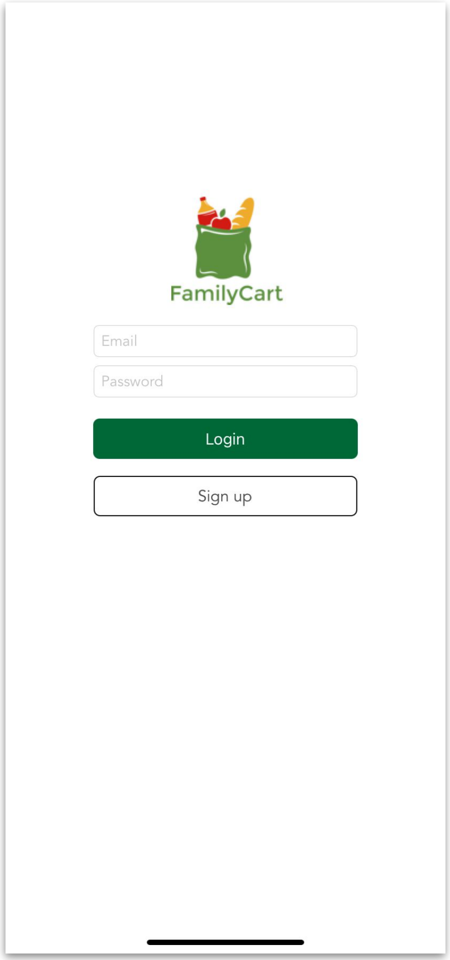
Functionalities

Login

Supports user authentication (***login*** & ***signup***)

Signup registers a new user (household member), and then logs in the new user

Login is for existing users



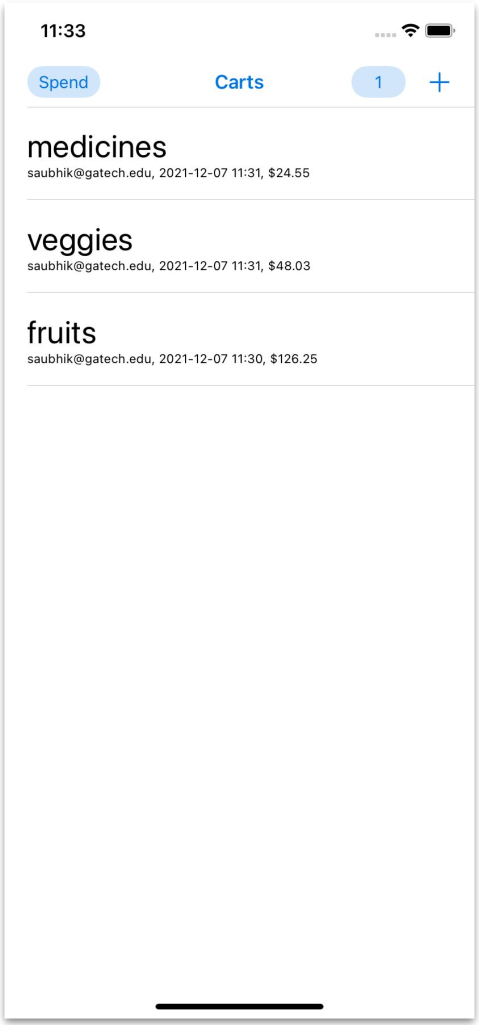
Carts

Displays a list of carts, along with **`addedByUser`**, **`addedOn`**, **`totalPrice`**

`Spend` leads to spend screen displaying simple ***spending stats***

Button on right with number shows **`onlineUsersCount`**, touching it leads to screen displaying list of ***users currently online***

+ button ***adds new cart***



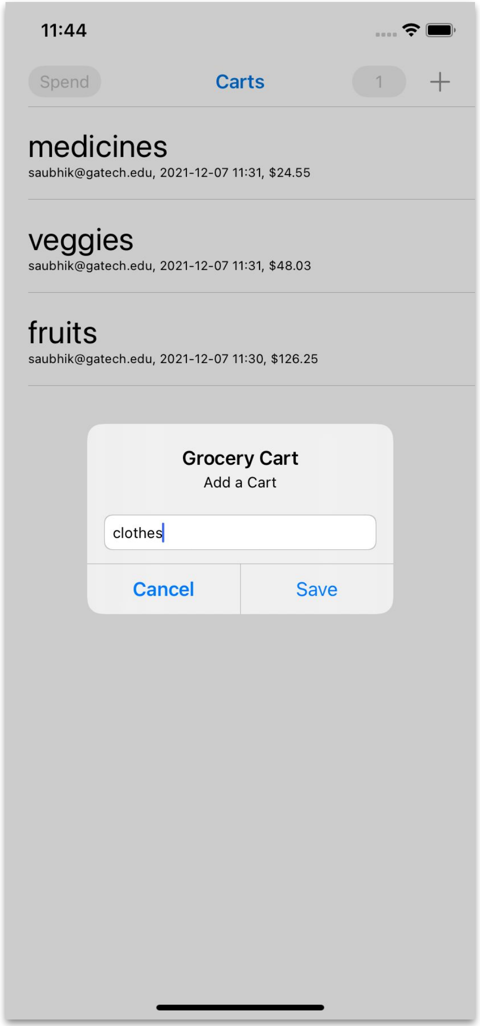
11:33

Spend Carts 1 +

medicines
saubhik@gatech.edu, 2021-12-07 11:31, $24.55

veggies
saubhik@gatech.edu, 2021-12-07 11:31, $48.03

fruits
saubhik@gatech.edu, 2021-12-07 11:30, $126.25



11:44

Spend Carts 1 +

medicines
saubhik@gatech.edu, 2021-12-07 11:31, $24.55

veggies
saubhik@gatech.edu, 2021-12-07 11:31, $48.03

fruits
saubhik@gatech.edu, 2021-12-07 11:30, $126.25

Grocery Cart
Add a Cart
clothes
Cancel Save

Items

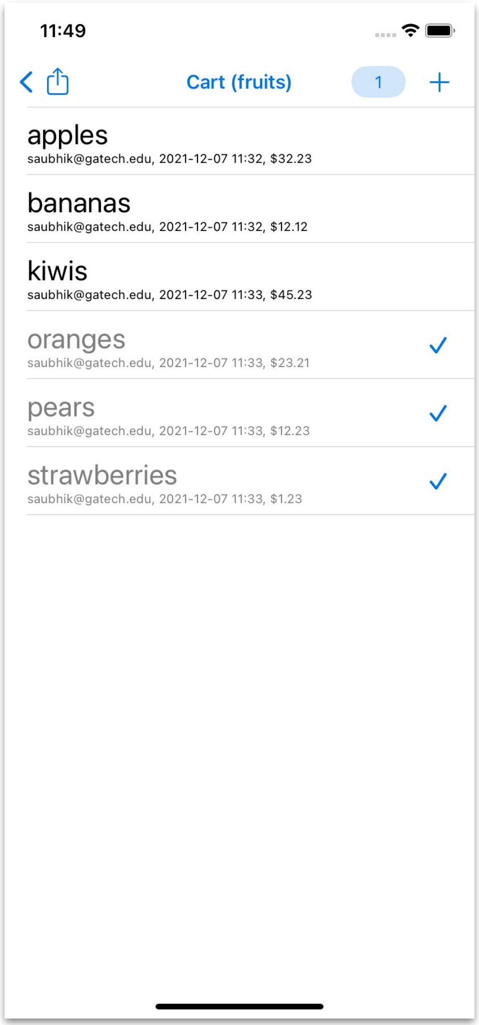
Displays list of items in chosen cart, along with **`addedByUser`**, **`addedOn`**, and **`price`**

+ adds a new item in the cart

Button on right with number shows **`onlineUsersCount`**, touching it leads to screen displaying list of ***users currently online***

Touching an item marks it as ***completed*** (grayed out with tick)

A cart is marked ***completed*** when all items are completed



11:49

Cart (fruits)

1

apples
saubhik@gatech.edu, 2021-12-07 11:32, $32.23

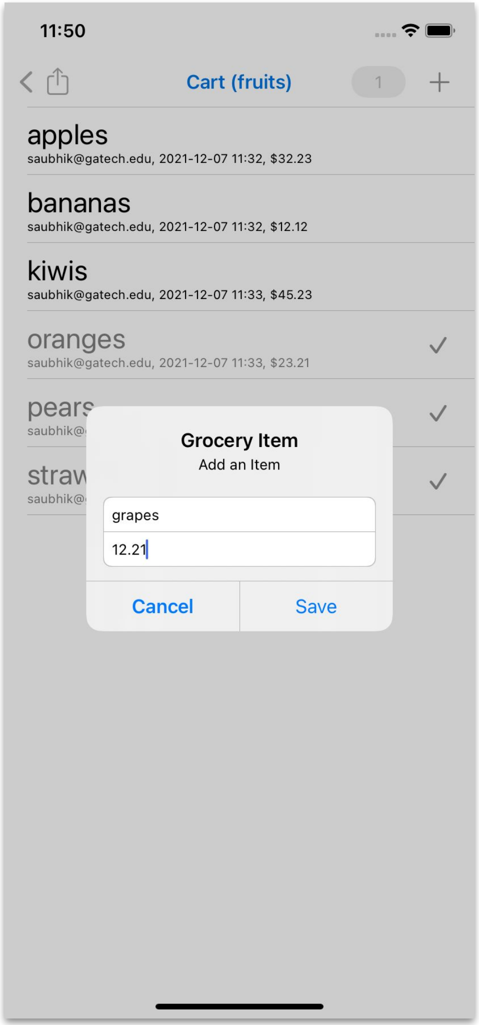
bananas
saubhik@gatech.edu, 2021-12-07 11:32, $12.12

kiwis
saubhik@gatech.edu, 2021-12-07 11:33, $45.23

oranges ✓
saubhik@gatech.edu, 2021-12-07 11:33, $23.21

pears ✓
saubhik@gatech.edu, 2021-12-07 11:33, $12.23

strawberries ✓
saubhik@gatech.edu, 2021-12-07 11:33, $1.23



11:50

Cart (fruits)

1

apples
saubhik@gatech.edu, 2021-12-07 11:32, $32.23

bananas
saubhik@gatech.edu, 2021-12-07 11:32, $12.12

kiwis
saubhik@gatech.edu, 2021-12-07 11:33, $45.23

oranges ✓
saubhik@gatech.edu, 2021-12-07 11:33, $23.21

pears ✓
saubhik@

straw ✓
saubhik@

Grocery Item
Add an Item

grapes

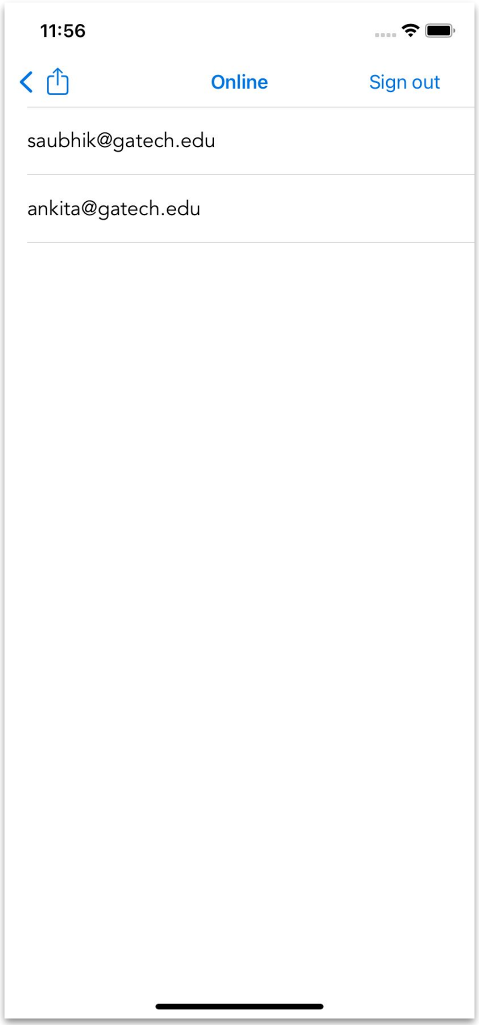
12.21

Online Users

Displays list of ***currently online users***

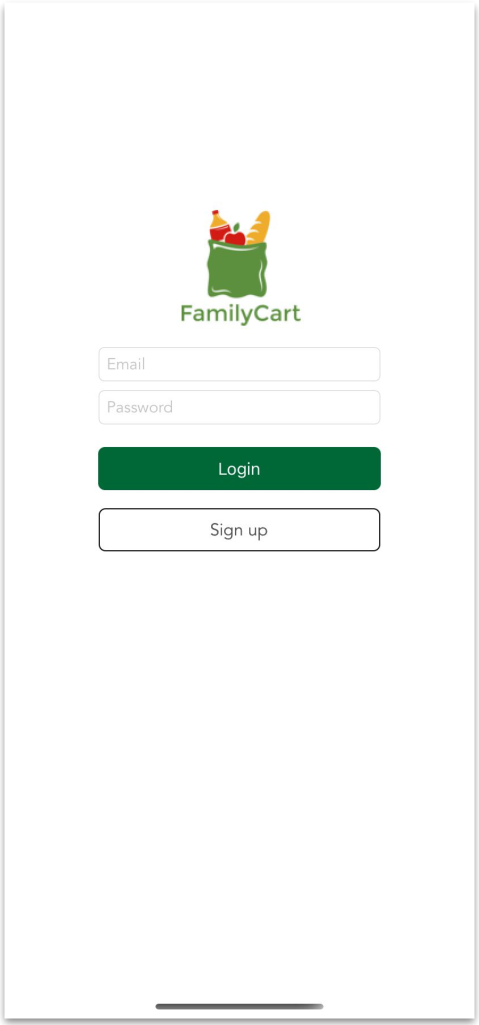
`Sign out` button signs you out of the application

`Back` button goes to previous screen



saubhik@gatech.edu

ankita@gatech.edu

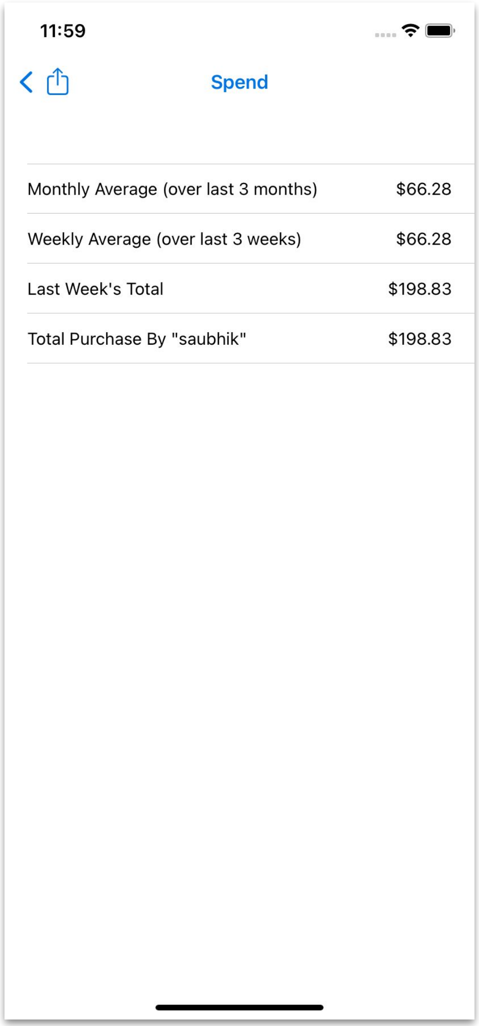


Spend

Displays simple ***spending stats***

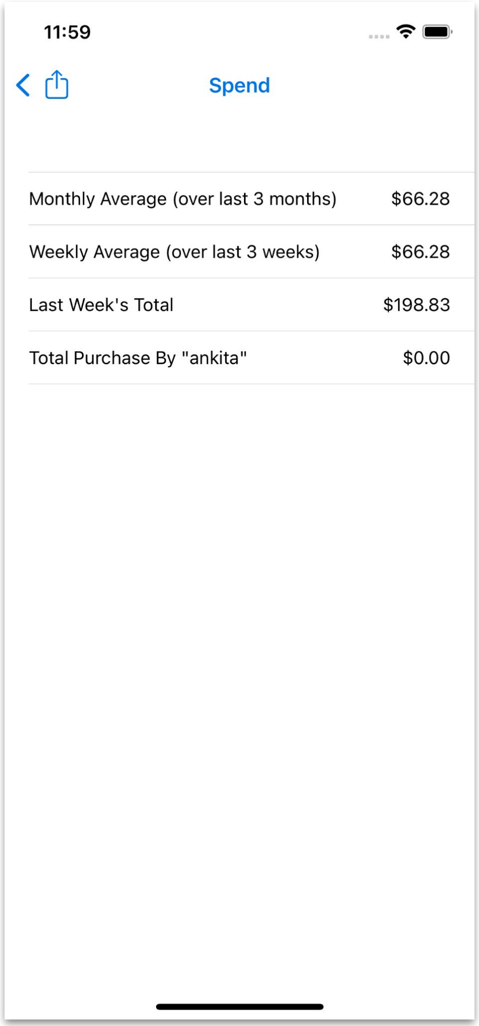
- ***average monthly spending***, averaged over last 3 months
- ***average weekly spending***, averaged over last 3 weeks
- ***total last week spending***
- ***total purchase made by current user***

`Back` button goes to previous screen


11:59

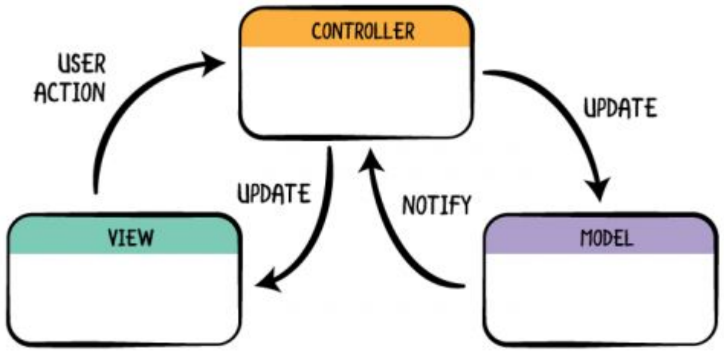
Spend

Monthly Average (over last 3 months)	$66.28
Weekly Average (over last 3 weeks)	$66.28
Last Week's Total	$198.83
Total Purchase By "saubhik"	$198.83


11:59

Spend

Monthly Average (over last 3 months)	$66.28
Weekly Average (over last 3 weeks)	$66.28
Last Week's Total	$198.83
Total Purchase By "ankita"	$0.00



Design

Figure source: https://www.raywenderlich.com/1000705-model-view-controller-mvc-in-ios-a-modern-approach

MVC Architecture

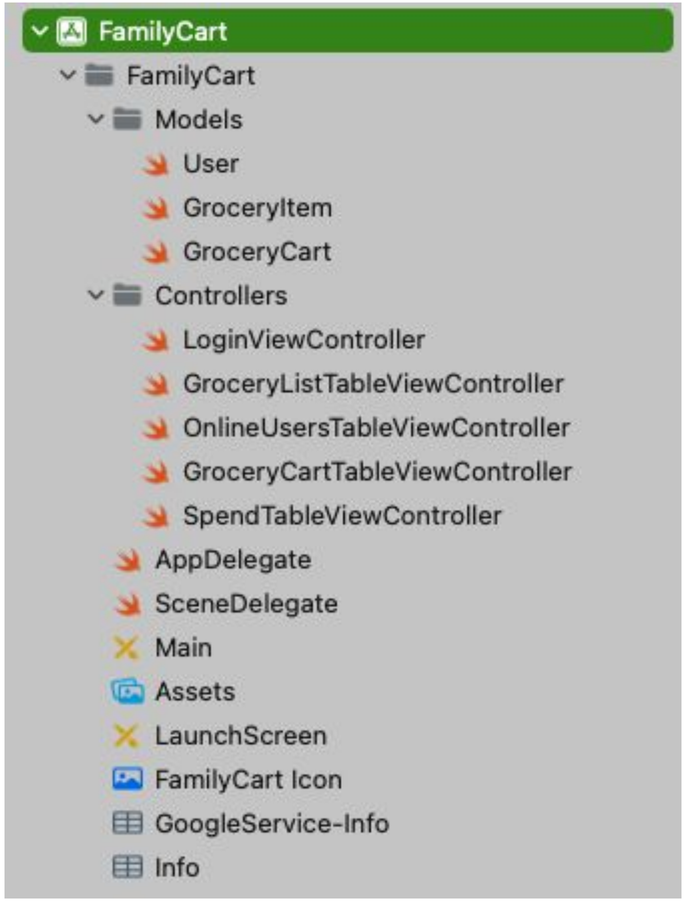
3 ***models***:

- User
- GroceryItem
- GroceryCart

5 ***view controllers*** for 5 views:

- login
- carts
- items
- online users
- spend stats

Intelligent ***controllers***, dumb ***models***

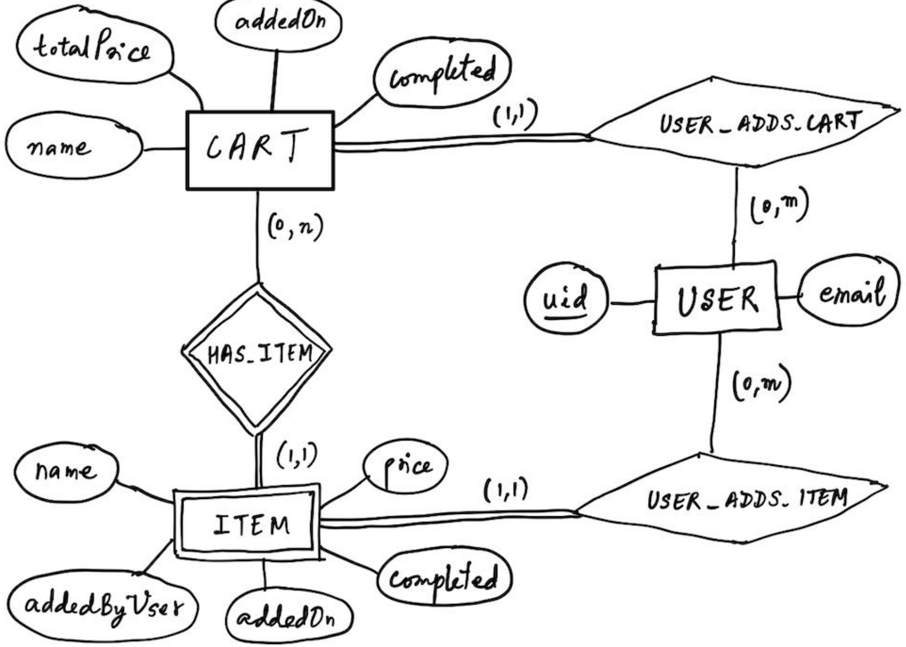


```
struct User {
  let uid: String
  let email: String
```

```
struct GroceryCart {
  let ref: DatabaseReference?
  let key: String
  let name: String
  let addedByUser: String
  let addedOn: Date
  var completed: Bool
  var totalPrice: Decimal
  var groceryItems: [GroceryItem]
```

```
struct GroceryItem {
  let ref: DatabaseReference?
  let key: String
  let name: String
  let addedByUser: String
  let price: Decimal
  let addedOn: Date
  var completed: Bool
```

ER Schema


totalPrice
addedOn
completed
name
CART
(1,1)
USER_ADDS_CART
(0,n)
(0,m)
uid
USER
email
HAS_ITEM
(0,m)
(1,1)
name
price
(1,1)
USER_ADDS_ITEM
ITEM
addedByUser
addedOn
completed

Challenges & Future

Challenges

Background (& interested) in *low-level systems development* (NetSys, `C`, `C++`, `Python`), tried ***mobile application development for first time***!

No prior background in ***Swift***, ***Xcode***, ***iOS***, ***NOSQL*** databases, ***frontend development***!

Focussed on getting a ***usable product*** out of the project, not necessarily feature-complete!

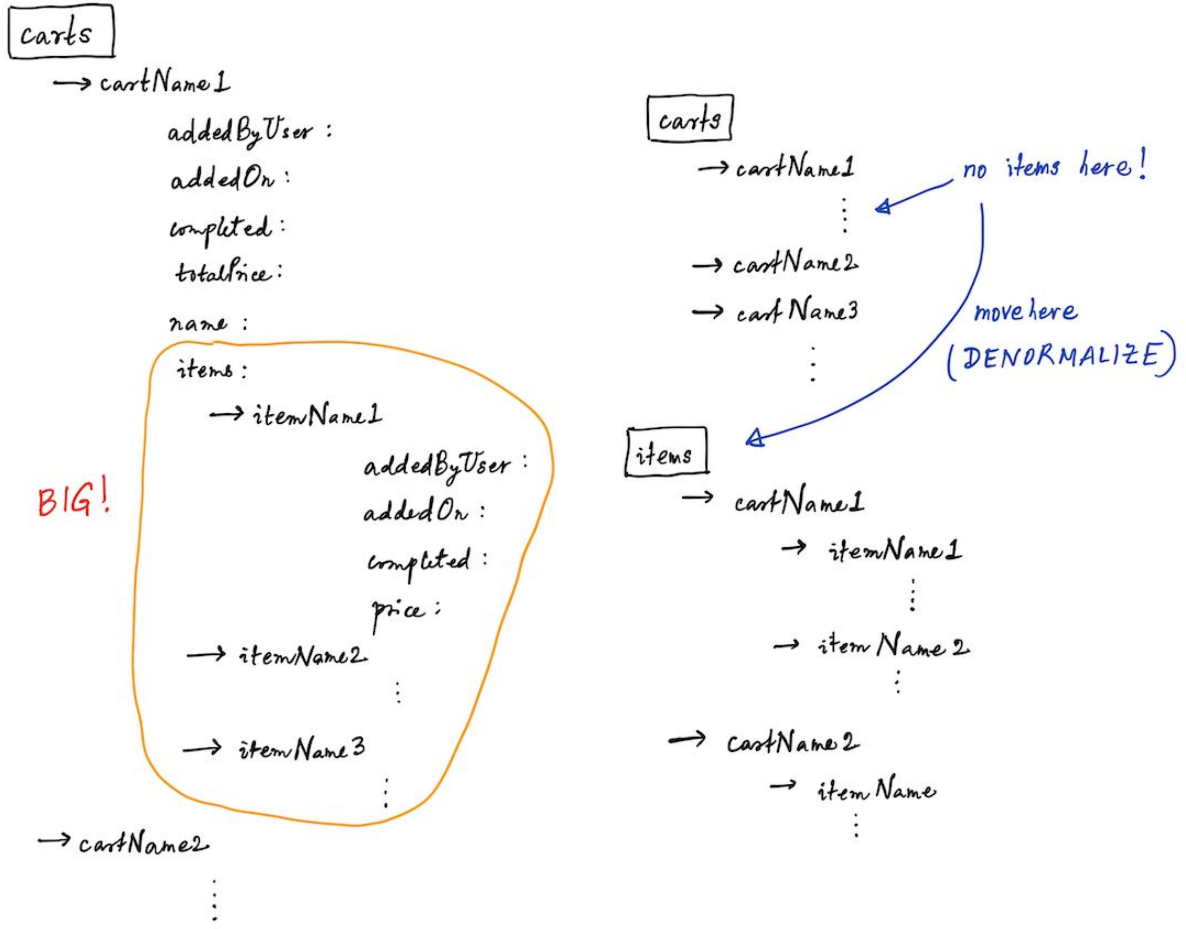
Familiar with ***PostgreSQL***, felt more constrained with available query APIs in ***Firebase***

Challenges & future

Emphasis on structuring data (***JSON tree***) in ***Firebase*** for efficient queries

Avoid ***nesting*** data, keep as ***flat*** as possible (***denormalization***)

Carts screen & spends screen can ***download hundreds of megabytes*** with current data structure! BAD!



https://firebase.google.com/docs/database/ios/structure-data

Future

Want to **`ORDER BY completed, addedOn DESC`**, but ***multiple orderings NOT possible*** in Firebase → ***restructure data!*** → split **`carts`** into **`incompleteCarts`** and **`completeCarts`**?

No notion of ***user accounts*** for supporting multiple households → **`account`** has **`carts`** → client should only download data for single account (***privacy***!)

More features

- **`purchasedByUser`** to **`cart`**, whoever completes last **`item`** in **`cart`**
- **`itemQuantity`** to **`item`**
- modify or remove **`item`**s or **`cart`**s