

# SUM System Design

---

---

---

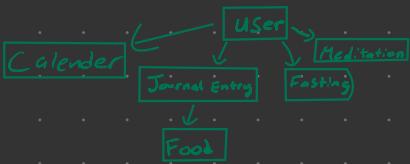
---





# Data Model

What is the relationship b/w Calender to (Fasting, Meditation, Journal)



User
id - PK
email - String
name - String
Membership_Status
weight
height

JournalEntry
id - PK
User_id - FK
date - Datetime
food_id - FK
EventID - FK

FoodEntry
id - PK
User_id
JournalEntryID

PK - Primary Key  
FK - Foreign Key

with this structure we can establish a many-to-one relationship between the Food table and Journal Entry Table

\* Fasting/Meditation can be one table (Activity/Event)

Event
id - PK
User_id - FK
start_time - TIME
End_time - TIME
Duration(hrs/mins)
notes - String(optional)
mood - String
Type (Meditation/Fasting)

Meal
ID
MealCategory:
FoodEntryID
00-to-2

\* Many-to-many Junction Table

MealInventoryItem
MealID
InventoryItemID

InventoryItem
ID
Macros
Calories
Description
Quantity

InventoryItemID  
needed b/c  
1. Data normalization

We can't store a list of InventoryItems in a single row of Meal Table.  
2. Query Performance

w/o junction table, Inventory Items association w/ particular meal req. full table scan

Videos
id - PK
videourl - String
title - String
description - String
category - String
likes - int
views - int
uploaded_date - time
thumbnail_url - String

Videos are very resource intensive  
utilize CDN

FoodHistory

We should also discuss the motivational short videos we want to implement, how we would use a video scrolling feed like (TikTok) for motivation

Meditation
id - PK
user_id - FK
date - Date
duration - TIME (mins)
mood - String (optional)

Calender
id - PK
date - TEXT
Fasting_id - FK
Meditation_id - FK
Food_journal_id - FK

Notes can be user writing how they feel during the fasting period

# SUM App System Design

System Design: Distributed Systems, Backend

Client is a Frontend UI  
Design Problem

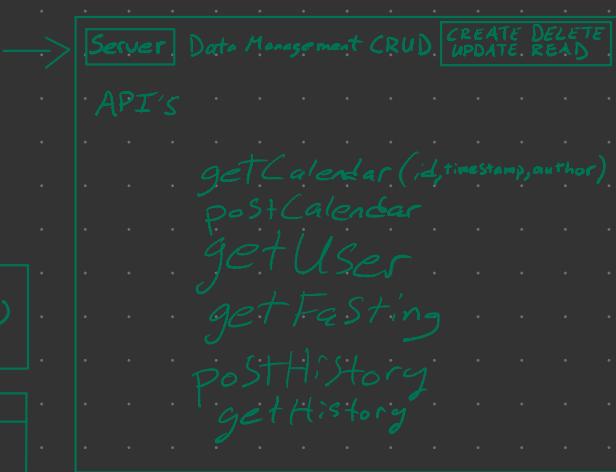
Customer → SERVER → Database



Frontend  
Mobile (Tablet/Phone)  
Web (Desktop)

## Required Functions

- Fasting Timer
- Food Diary
- Food History
- Meditation Timer
- Calendar
- Offline Mode
- Food Search
- Video feed (for fasting)



## Mobile Specific Problem

1. Limited Resources: Internet traffic, battery usage, memory
  2. High Level Architecture
  3. Design Patterns
  4. UI Architecture
  5. Backend
  6. Data Storage
  7. App Specific
- Solution needs to be organized to reduce usage of these resources.

SQL DB Local Storage  
SQLite

Author is user Foreign Key

## Calendar

ID	Timestamp	Author
1.	DATE	User
2		
3		
4		
5		

User
ID
Name

History
ID
Food





