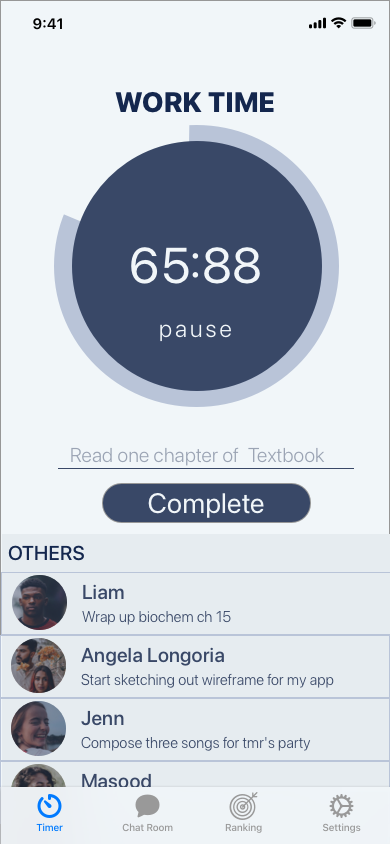
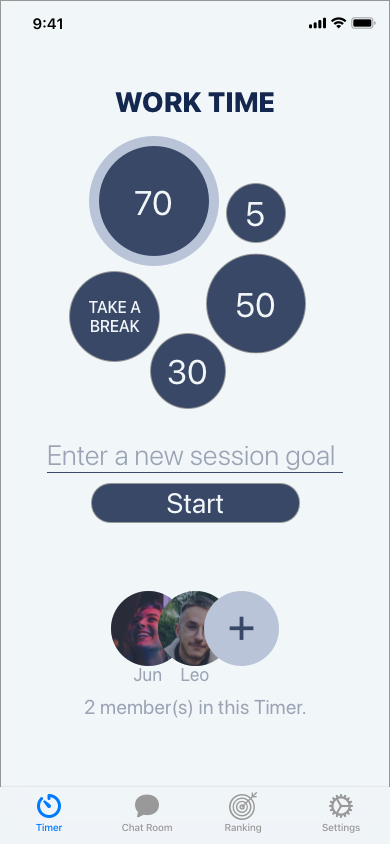
TimeShare

TimeShare is an app where friends can share a pomodoro timer to work and take breaks together . Before starting a session, each person will be asked to enter a session goal so we can hold each other accountable during work. During a session, you can see each other’s goals as a way to hold each other accountable.

TODO:

* Timer Screen
  + Choose a timer
  + Share a timer
  + Choose a timer
* Navigation controller screen to add another person
* Chatroom Screen where friends can talk during work and break time
* Ranking Screen to compare study hours and trends
  + You are assigned a study role depending on the hours you’ve spent this month
  + You can look at the study trends (hour for each week, month and year)
  + You can look at the other role
* Settings Screen to configure account and more personalization
  + Configure time zone
  + Configure account information
    - Configure Avatar, Email, Password



# 

# Feature List

## Minimum Requirements

* Show history of each break, and each study session
* Share the timer with another person without having to log in
  + Have another person enter a secret code to join in on the same room
* Show all members in the room
* Show time zone for each member

## Standard Requirements

* Account for each person containing email
  + Firebase to handled account log in
* Database storing the current timer so that when the app is close, the timer continue to run
* Group Chat
* Support Turning On Camera

## Stretch Requirements

* Stats that compares ranking of each account
  + Fancy Bar graph comparing between accounts
  + Fancy line graph looking at trend of an account
* Supporting gifs and custom images on Group Chat
* Support Turning on Screen Share
* Dark - Light Mode

# 

# Color Scheme



f1f6f9, 14274e, 394867, 9ba4b4 (light mode)



9ba4b4, 394867, 14274e, f1f6f9 (dark mode)

## 

# Timeshare Database Schema:

Key Formats: Primary Key

**Foreign Key**

*Candidate Key*

Timer (timerID: String, start: Date, interval: Double, pause: Boolean, reset: Boolean, stop: Boolean)

Room (roomID: String, capacity: Integer, **timerID**: String, inSession: Boolean)

* timerID: references Timer

User (userID: integer, **timerID**: String, **roomID**: String, photo: Image, timeZone: String)

* timerID references Timer
* roomID references Room
* [timeZone format](https://stackoverflow.com/a/33465436)

History (histID: String, **userID**: String, dateAdded: Date, hours: Double, type: String)

* userID references User
* type: Can be either “Break” or “Session” (TODO not safe using string)

Session (sessionID: String, **userID**: String, **histID**: String)

* userID references User
* histID references History

Goal (goalID: String, **userID**: String, **sessionID**: String, completed: Boolean)

* userID: referenced User
* sessionID: references Session

Break (breakID: String, **userID**: String, histID: String)

* userID references User
* histID references Break

## 

## Functional Dependencies

Key Formats: Primary Key

**Foreign Key**

Dependencies: Non-Trivial Dependencies

1. Timer
   1. timerID → start
2. Room
   1. roomID → **TimerID**
3. User
   1. userID → photo, **roomID**
4. History
   1. histID -> **userID**, dateAdded, hours, type
5. Session
   1. sessionID, userID → histID
6. Goals
   1. goalID → userID, sessionID
7. Break
   1. breakID → userID, histID

Questions

# Cloudkit+CoreData, Realm or Firebase?

Given the timer is synced between not just personal devices, between devices from multiple people. I wonder if CloudKit is still the best tool to share a timer and other information between people.

I think I will not use CloudKit since it limits the possibility for the service of this app to expand beyond iOS. For simplicity, I think I will go with **Firebase**. I thought about maybe using Realm Cloud but it costs at least 30 dollars a month in advance with no free tiers :(.