Getter Journal 1: Karthik Ponnapalli

App Idea: The Getter app is a to-do list app with a spin: assigning tasks based on the start date rather than end date. The app intends to help students with task management and proactivity.

GitHub: https://github.com/karthikponnapalli/Getter-Repo.git

MVP Outline (FE/BE):

Things to do before "sprint":

Develop/Finish Flow Charts
Develop/Finish wireframes
Learn swift
Understand APIs

Key:

"Assignments" refer to all assignments (assignments, quiz, tests)

Features to be included in MVP:

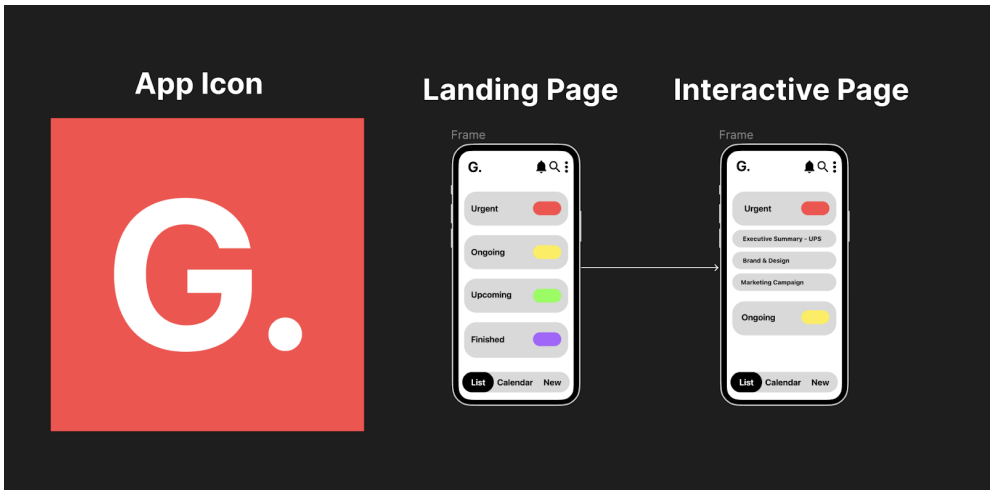
1. Add assignments
2. Remove assignments
3. Sort assignments by priority (Urgent, Ongoing, Upcoming, Finished)
    a. Start Date Calculation
    b. Data Structures & Algorithms
4. List View/Calendar View
5. Account features?
6. Features on wireframe
    a. Menu/Settings
    b. Search (for assignments)
    c. Notification Bell

UI/UX:

Backend Code:

1. How do/should we add and store assignments?
2. How do/should we remove assignments?
3. How can we sort assignments
    a. By start date, sort into priority buckets
4. How can we bring a calendar into our app
    a. (canvas or google calendar API)
    b. How can we upload start - end date timeline to calendar

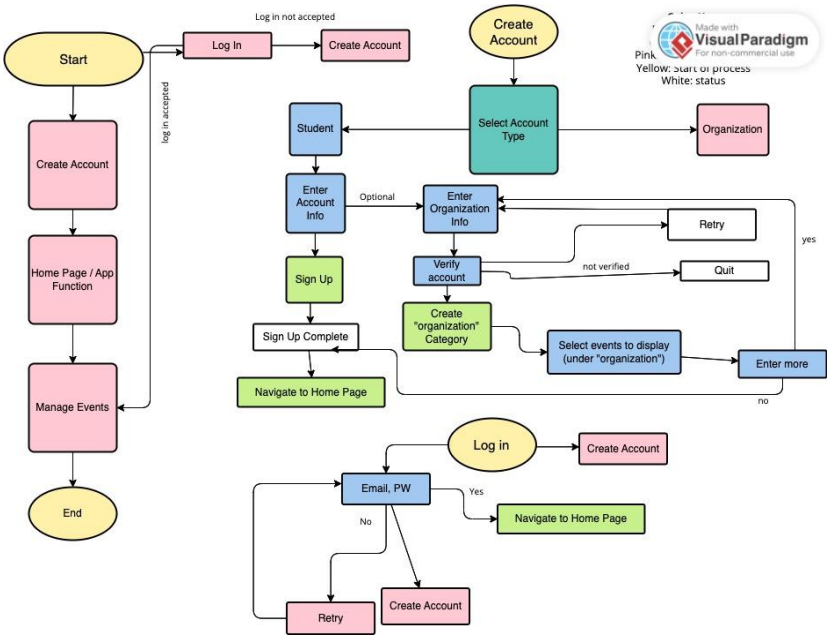
App Icon & Landing Page:

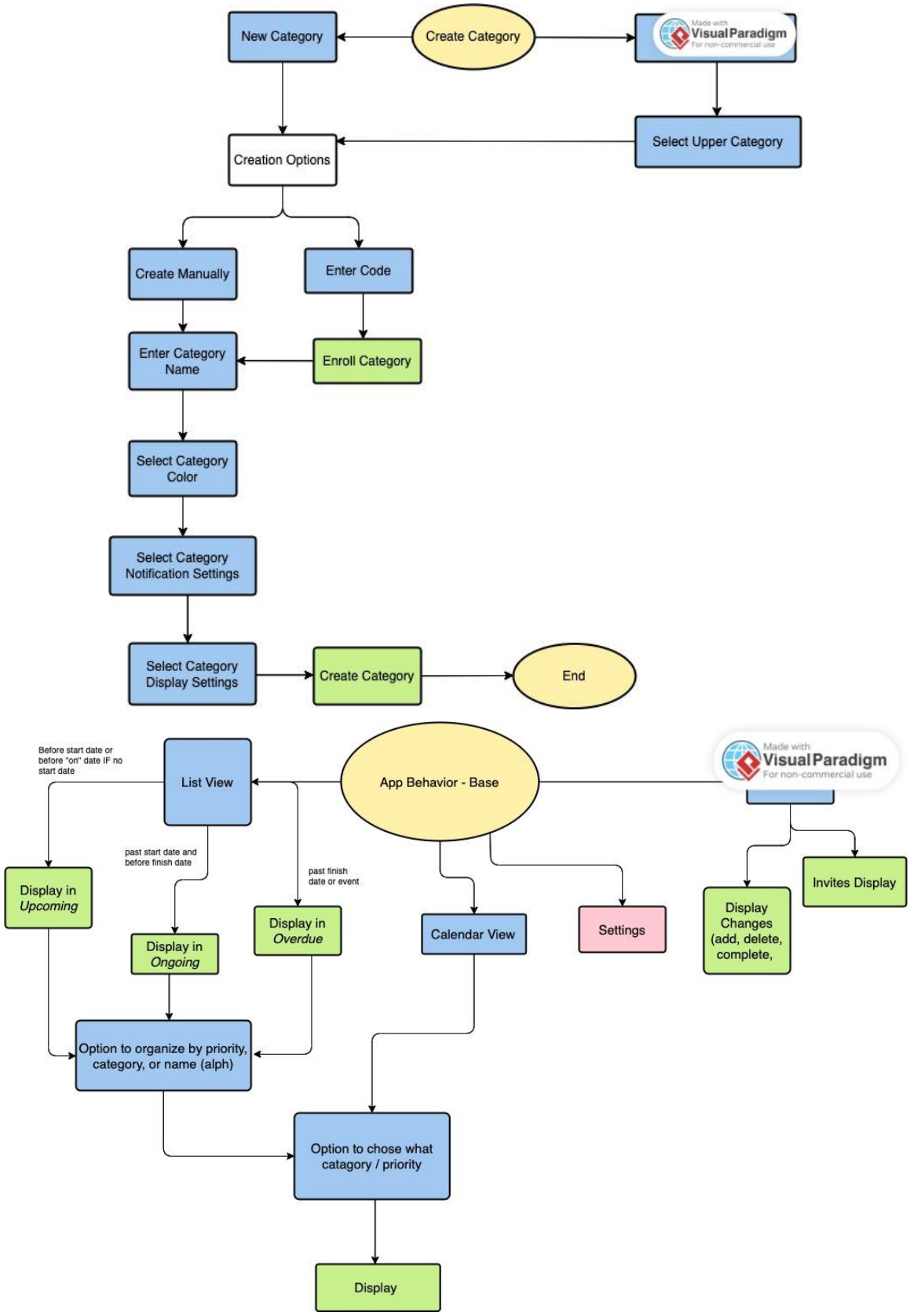


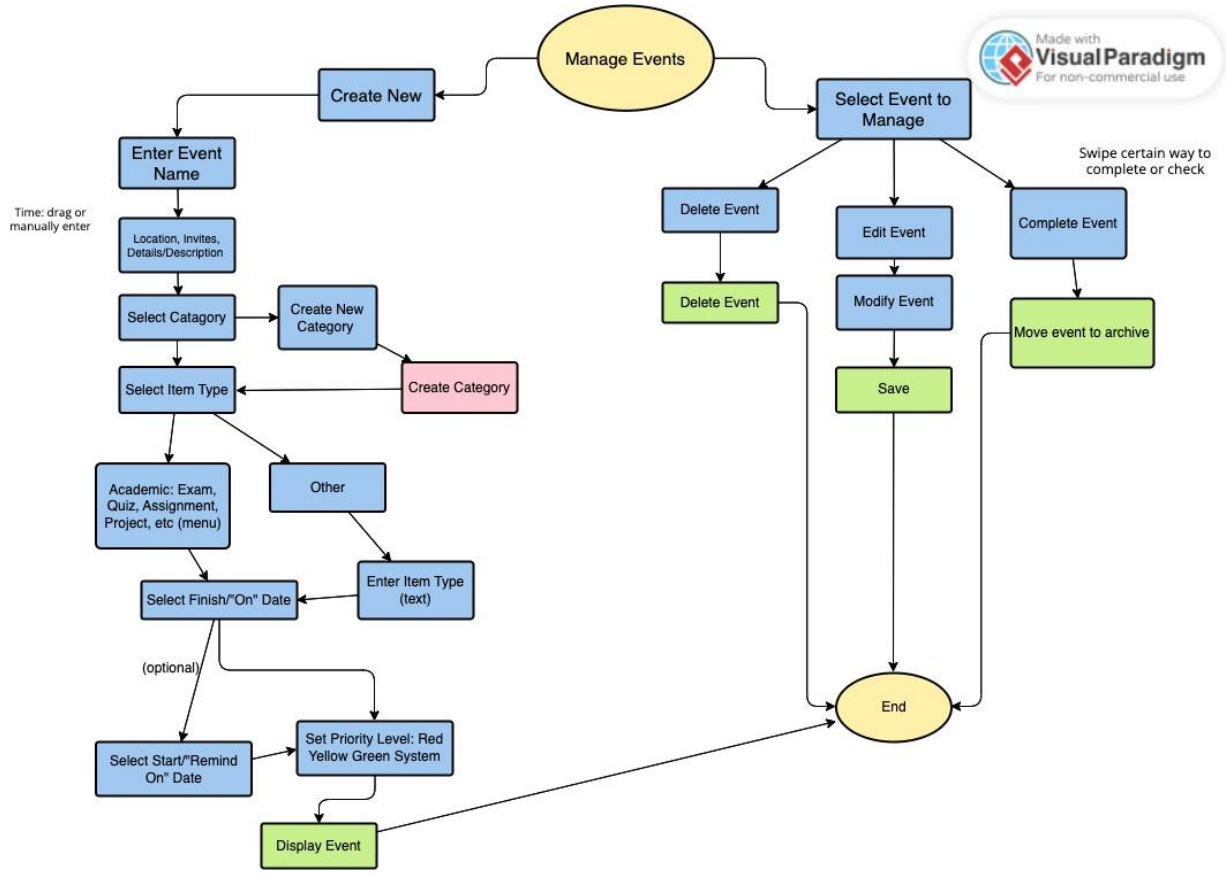
We wanted a simple yet aesthetic app icon that will invite users to download the app. The “G” is the first letter of the app name, and the period, which traditionally represents the end of a sentence, is meant to indicate the completion of tasks. Our landing page is intuitive and easy to follow. We wanted our users to be able to familiarize themselves with the app quickly as complicated apps may deter potential users.

Tasks fall under four bins: urgent, ongoing, upcoming, and finished. There are also three views: list, calendar, and new. The list view has the four bins. The calendar shows a calendar view of your assignments with the start dates and due dates. The new button allows users to create a new event and input information such as start date, new date, and urgency.

User Flows:


Start
Log In
Log in not accepted
Create Account
log in accepted
Create Account
Home Page / App Function
Manage Events
End
Create Account
Made with VisualParadigm
For non-commercial use
Pink:
Yellow: start of process
White: status
Select Account Type
Student
Organization
Enter Account Info
Optional
Enter Organization Info
Retry
yes
Verify account
not verified
Quit
Sign Up
Create "organization" Category
Sign Up Complete
Select events to display (under "organization")
Enter more
Navigate to Home Page
no
Log in
Create Account
Email, PW
Yes
Navigate to Home Page
No
Retry
Create Account


New Category
Create Category
Select Upper Category
Creation Options
Create Manually
Enter Code
Enter Category Name
Enroll Category
Select Category Color
Select Category Notification Settings
Select Category Display Settings
Create Category
End
Before start date or before "on" date IF no start date
List View
App Behavior - Base
past start date and before finish date
past finish date or event
Display in Upcoming
Display in Ongoing
Display in Overdue
Calendar View
Settings
Display Changes (add, delete, complete,
Invites Display
Option to organize by priority, category, or name (alph)
Option to chose what catagory / priority
Display



Development/Tasks:

- Developing a timeline to complete tasks
- Currently creating wireframes on Figma using the user flows
- Creating initial classes/structures for app in swift
- User Testing the idea (in marketing class)

Tic Tac Toe Game:

- Coded a tic tac toe game
- You can play against different opponents
    - Bot
    - Two players share the screen
    - Challenge a friend
- Theres a start button
- Declare what piece before you play
- Theres a grid
- Different assets for X and O
- Winner/Loser/Draw screen
- Choose player to start
- End game function

EdTech Platform:

- Uses JSON files as a way to pull data into the platform
- Created Quizzes and Lessons

Other Projects (Ideathon and Personal):

- Wire Framed Ideathon app on figma
    - Created an employee dashboard with 5 different views with different functionality
    - Created a user dash board with different functionality and customizations based on surveys
- Learned to use figma to code and figma to swift to help with the coding process
- Learned how to properly differentiate between structs, classes, enums

Helpful Links:
https://www.youtube.com/watch?v=Vqo36o9fSMM
https://docs.swift.org/swift-book/documentation/the-swift-programming-language/thebasics/
https://blog.prototypr.io/7-rules-for-creating-visually-aesthetic-ui-6b8784fced9a