ZipApp - Orange Team
Architectural Spike Report

Authors
Trevor Aupperle
Benjamin Fisk
Camden Davis
Mohab Yousef

COMP 4710 Senior Design
Auburn University Samuel Ginn College of Engineering

January 29, 2024

1 - System Metaphor - Trevor

The system metaphor for this app is similar to a digital “gameday caddy”. Just as a caddy in golf helps players by navigating them around the course while carrying their clubs, this app is a personal navigator through campus on football game days. With a friendly and intuitive interface, users can request rides from golf carts in the area to reach their desired destinations on campus with ease. This creates a quick, efficient, and hassle-free environment for fans from all over.

2 - Cycle Intent - Ben

The intent for this cycle is for all of our team members to gain a thorough understanding of the code base, and to establish working development environments to facilitate productivity throughout the semester. We will be exploring the possibility of having some developers target the Android operating system to verify whether or not the application is able to be run on the OS. Additionally, our team further intends to explore our options in bringing updates to the codebase, to ensure that the application can safely interface with current software versions. Finally, we plan to use any remaining time in this cycle to enumerate the tasks required to implement the features that our sponsor is interested in having added to their application.

3 - Future Work - Camden

In future cycles, we intend to focus on implementing the features that the sponsor has given to us. These features include the following: Bringing directions in the form of google maps integration to the drivers of golf carts; Bringing real time notifications and updates to the in-app map for the user; Bringing a feedback and rating system to the app for users to rate their drivers; Bringing a record system for past trips to both the driver and rider sides of the app.

4 - User Stories - Trevor

Infrastructure

1.0 Flutter Migration
Summary: As a user, I need to be able to use the app to work as intended with no security flaws.
Description: The current Flutter version used in the application from previous teams is Flutter 1.22.5. This is a major concern since Flutter has since released 2 major version updates which include null-safety checking, sunsetting of old iOS versions (iOS 9 and iOS 10), and performance/security improvements. All of these are necessary to ensure the application works as intended and improves the security vulnerability risks.
Planned Hours: **8**

Planned Hours this Cycle: **0**
Actual Hours: **12**
Actual Hours this Cycle: **0**
Coder Names: **Trevor Aupperle**
Tester Names: **Entire Team**
Review Names:
Status: COMPLETE

1.2 Operating Systems Check
Summary: As a user, I need to be able to use the app on the most recent versions of iOS and Android operating systems.
Description: After migrating to Flutter 3.0, we need to ensure we can run the app on simulators for different operating systems.
Planned Hours: **1**
Planned Hours this Cycle: **1**
Actual Hours:
Actual Hours this Cycle:
Coder Names: **Ben Fisk, Camden Davis, Mohab Yousef**
Tester Names:
Review Names:
Status: IN PROGRESS

1.3 flutter_facebook_login Package Migration
Summary: As a user, I need to be able to login to the app using my Facebook profile.
Description: The Flutter package, flutter_facebook_login, is no longer compatible with Flutter

3.0 and has not been updated in over 4 years. We need to find a new package that is capable of handling facebook sign-ins (we should probably use firebase_ui_oauth_facebook since it is specifically used in conjunction with Firebase and is built by Google).
Planned Hours: **6**
Planned Hours this Cycle: **6**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

1.4 geoflutterfire Package Migration to geoflutterfire2
Summary: As a user, I need specific geolocation features to work so that I can decide where to go and figure out where I am being picked up.
Description: The current geoflutterfire package is incompatible with newer versions of Firebase packages. We need to migrate to the more updated geoflutterfire2 package to handle the dependency issues. To do this, we will need to remove the package flutter_google_places and will need to find a replacement or build our own solution.
Planned Hours: **6**
Planned Hours this Cycle: **6**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

1.5 Find an Alternative to unicorndial Package

Summary: As a user, I need an intuitive and friendly user interface to interact with.
Description: There is a Flutter package, unicorndial, that is currently being used as a user interface component. However, it has not been updated in 5 years and no longer works with the new Flutter version. We need to find an alternative or build our own component.
Planned Hours: **4**
Planned Hours this Cycle: **4**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

1.6 stripe_payment Package Migration to flutter_stripe
Summary: As a user, I need a reliable purchasing system to make purchases on the app for the rides I request.
Description: Currently, the app uses the package stripe_payment to handle payments with the payment provider Stripe. Stripe has migrated to a newly updated package flutter_stripe. Changes need to be made in the app to handle the new package.
Planned Hours: **8**
Planned Hours this Cycle: **8**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

UI/UX

2.0 Figma Mockups
Summary: As a user, I need an intuitive and friendly interface to interact with.
Description: Currently, the user interface for the app is simply bad. We need to draw up high-fidelity prototypes in Figma to create a better UI/UX system for users.
Planned Hours: **12-14**
Planned Hours this Cycle: **12-14**
Actual Hours:
Actual Hours this Cycle:
Coder Names: **Trevor Aupperle**
Tester Names:
Review Names:
Status: **NOT STARTED**

2.1 Implement UI Changes
Summary: As a user, I need an intuitive and friendly interface to interact with.
Description: After mocking up prototypes in Figma and getting them approved by the sponsor, we need to implement the features in the code.
Planned Hours: **12**
Planned Hours this Cycle: **0**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

Functionality

3.0 Driver Directions Research
Summary: As a user (driver), I need to be able to see the best route to take customers to their desired destination.
Description: We need to limit driver routes to streets that can accommodate golf carts legally (35 MPH or less) and also be able to show routes that will likely be “blocked” on game days for vehicles. We will have credentials to get through “blocked” roads. Need to take into account cost algorithms when researching. Complete research of how we can accomplish this.
Planned Hours: **12**
Planned Hours this Cycle: **12**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

3.1 Driver Directions Implementation
Summary: As a user (driver), I need to be able to see the best route to take customers to their desired destination.
Description: We need to limit driver routes to streets that can accommodate golf carts legally (35 MPH or less) and also be able to show routes that will likely be “blocked” on game days for vehicles. We will have credentials to get through “blocked” roads. Need to take into account cost algorithms when researching. Implementation of research conclusions from user story 3.0
Planned Hours: **12 - 16**
Planned Hours this Cycle: **0**
Actual Hours:
Actual Hours this Cycle:

Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

3.2 Rider ‘On the Way’ Notification
Summary: As a user, I would like a notification about when my driver is going to arrive.
Description: Add a notification for the rider when the driver is on the way to pick them up with an estimated arrival time. Riders should see where the driver is while they are en route.
Planned Hours: **8**
Planned Hours this Cycle: **0**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

3.3 Rider Status
Summary: As a user (rider), I would like to be able to see where I am at in my ride on a map as the drive is occurring.
Description: Once a ride has begun, the rider should see the golf cart location on a map as it moves along the drive.
Planned Hours: **6**
Planned Hours this Cycle: **0**
Actual Hours:
Actual Hours this Cycle:

Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

3.4 Rider Recap Screen
Summary: As a user (rider), I would like to see a recap of the ride I just took.
Description: Once a ride is over, a screen should be shown to the rider that summarizes their trip and costs. It should also prompt the user for a rating.
Planned Hours: **8**
Planned Hours this Cycle: **0**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

3.5 Drivers Near Me
Summary: As a user (rider), I would like to see where the nearest drivers are around my current location.
Description: Riders should be able to see current locations of carts around them on a map.
Planned Hours: **6**
Planned Hours this Cycle: **0**
Actual Hours:
Actual Hours this Cycle:
Coder Names:

Tester Names:
Review Names:
Status: **NOT STARTED**

3.6 Feedback/Rating System
Summary: As a user, I need to be able to give feedback and rate both driver and riders after interactions with them.
Description: Implement a 5-Star rating system.
Planned Hours: **10**
Planned Hours this Cycle: **0**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:
Review Names:
Status: **NOT STARTED**

3.7 Display Past Trips
Summary: As a user, I would like to be able to see all of my previous trips as both a rider and a driver.
Description: Create a drivers past trips log and a riders past trips log.
Planned Hours: **12**
Planned Hours this Cycle: **0**
Actual Hours:
Actual Hours this Cycle:
Coder Names:
Tester Names:

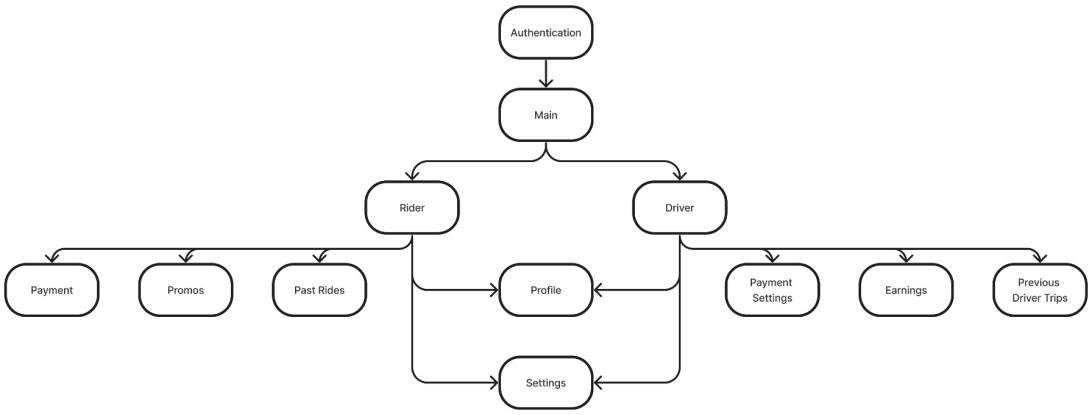
Review Names:
Status: **NOT STARTED**

5 - Design Documentation - Ben

Architecture

The Zip application is designed to act as a service to bring college football fans to and from stadiums on game days via golf carts. This idea stems from the desire of fans to have a method of transportation into crowded areas where cars are not permitted to drive. The application hinges on user-profile based interaction. Each user will have their own profile for either reserving rides or offering driving services. A user that logs into their rider profile will have access to reserving and paying for a ride, their ride history, and promotions going on at the current time. A user that logs into their driver profile will have access to their payment settings, their earnings, and previous trips they have driven. Both riders and drivers will have access to their general profile and settings.

See below a diagram depicting the flow of a user's action inside the Zip app.



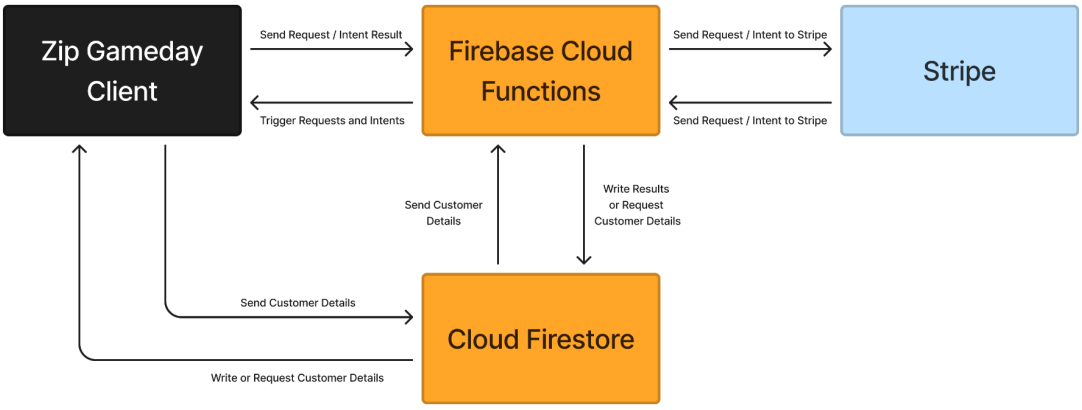
Structure

The application is built using the Flutter framework. Flutter was built by Google to leverage the Dart programming language to build applications. The combination of Dart and Flutter is designed to allow applications to be installed and run on both the IOS and Android operating systems.

Interfaces

The application will make use of Google Firebase and Cloud Firestore for database, authentication, and cloud services. It will also be using Stripe to handle transactions between riders and drivers. These services necessitate an internet connection for the app's functionality.

See below a diagram depicting the interactions between the Zip application, the Firebase cloud services, and the Stripe transaction service.



Justification of Decisions

The decision was made to use Flutter, because applications made with Flutter can be run on IOS and Android. The decision was made to use Google Firebase and Cloud Firestore because the database and authentication services they provide would be difficult to recreate from scratch given the scope of this project. The decision was made to use Stripe for similar reasons, in addition to the concern of security as it pertains to conducting monetary transactions.

Assumptions and Other Dependencies

This application depends heavily on an internet connection for all aspects of its operation. From the user's login to the reservation of rides to the exchange of payment, the app will not function without being connected to the internet.

6 - Management Plan - Mohab

For the management of our project, we will be using Git for version control. We will also be in close contact with the Zipapp blue team, to ensure that no work is duplicated and no conflicts arise.

a. Task Assignments

We will be using Jira to communicate and manage tasks for the team. We will also be communicating via email and Groupme for meetings and other non-technical purposes. We plan on approaching the tasks for this project as a team to begin with, and then assigning tasks to individuals as we see fit based on the ease with which the task can be completed. This will allow everyone on the team to gain an understanding of each task and then make an informed decision on which and how many people should work on that task.

b. Development Schedule

Our team will be meeting weekly to discuss our progress, and we will have weekly or biweekly meetings with our sponsor, depending on the sponsor's wishes. These frequent team meetings will allow our team to stay on the same page, and also to move on to new tasks as old ones are completed. The meetings with our sponsor will allow us to keep them up to date, and ensure that our development of the application aligns with their vision.

7 - Risk Mitigation - Mohab

Our plan to mitigate risk for this project involves several strategies. The first of which is to maintain high levels of communication within our own team and between the orange and blue teams. This will ensure that the orange and blue teams are not completing overlapping tasks, and that all aspects of the application will mesh correctly as the two teams are working. Another strategy that we will be following involves ensuring that all technologies used in the project are up-to-date and secure. The application that we are developing involves user data and payment transactions, both of which require a high level of security to maintain the functionality of the application and also to maintain user trust. By paying close attention to user login information, profile data, and payment authentication, we can guarantee the integrity of the app.

8 - Lessons/Challenges - Camden

One of the main challenges we have faced so far is getting the application in its current state (from the previous team's code base) to simulate correctly. We quickly figured out the infrastructure for the application was outdated and poorly maintained. Our decision was to upgrade/fix the infrastructure of the application before attempting to make any more features. In making this decision we learned a lot about the migration process to newer versions of packages and SDKs. We also learned through this process that it is better to upgrade to major versions quicker rather than later. Because the other teams before us put it off, we were forced to make breaking changes that drastically increased the time it will take to fix the app.

Throughout the code base we found many instances of code that was poorly written; commented out lines of unused code; print statements that print critical information about the state of the app to the console; API keys directly in the code rather than in a .env file; and code that was completely unused. This is unacceptable in software development and it taught us that we need to implement a plan to have code reviews before pushing code to the code base. Therefore, we have decided that code cannot be pushed to the code base without at least one person (preferably two people) reviewing the code. This will ensure the code base is maintained properly, has minimal security risks, and is easily readable for future development.

9 - Meeting Minutes - Ben

Date: 01/25/2024

Time: 5:30 PM - 6:30 PM
Attendees: Trevor Aupperle, Ben Fisk, Camden Davis, Mohab Yousef, Russell Anderson, Paxton Delamar

1. Introductions - we all introduced ourselves including our background and experience.
2. Business Idea - Russell explained where the business idea originated (i.e. working with the football team to drive recruiters around game days on a golf cart)
3. 4 Main Tasks - Russell described the 4 main tasks that he and Paxton would like to see accomplished this semester.
    a. Driver Directions
    b. Rider Side Visuals
    c. Feedback/Ratings
    d. Display Past Trips
4. Infrastructure Concerns - Trevor brought up concerns about the current state of the project's infrastructure being outdated and incompatible with current operating systems (iOS and Android). Both teams (orange and blue) had been unable to get the app simulated in its current state. Therefore, we needed to spend time updating the infrastructure packages. Russell understood and agreed that was important to fix before moving forward.
5. Questions - We shortly discussed some general questions before ending the meeting.

Date: 01/26/2024
Time: 3:30 PM - 5:00 PM
Attendees: Trevor Aupperle, Ben Fisk, Camden Davis, Mohab Yousef, Nico Marthe, Zach Grindle, Jordyn Lewis

Both the Orange and Blue team for ZipApp met to discuss user stories and how we were going to communicate with each other during the semester. We set processes in plan to avoid confusion about how to merge code when working on the same codebase (i.e we will be doing code reviews in GitHub).

10 - Sponsor Approval - Trevor

Request: Email sent to Russell the morning of 01/29/2024

Sponsor Virtual Signature Approval:
(i.e. *John Doe, signed 01/01/2024*)

Russell Anderson, signed 01/29/2024