Sprint 3: API & Prototyping

Outrun

Mohona Sengupta and Nupur Maheshwari

67-442 MOBILE IOS | 10.24.2019

List of Features: V1 & V2

V1

- The User can input a number of minutes and severity, and Outrun will create a route for their walk
- The User can input a distance and severity, and Outrun will create a route for their walk
- During the walk: AR vision in the camera to see three game features: zombie bird, zombie human, and first aid kit
- During the walk: Outrun will give user directions on where to go
- During the walk: Ability to gain or lose points through phone hardware
- After the walk: Ability to see birds/zombies killed at the end
- After the walk: Ability to see final points in the end
- Information section
- TENTATIVE (might not be in our MVP for V1 but we want to try): Ability to see past routes and select those
- TENTATIVE: See the final points along with graph of points achieved every time you took the route

V2

- User gets 5 routes to their specification during route creation, and can choose one
- Store feature, where the User can buy clothes/armor/upgrades with their points

Why We're not Using an Internal API

With the help of our mentor and Prof H, we outlined our options when it comes to storing data: using *core data* (or the plist), using an *internal* API, or using *GameCenter* (having the user create a profile and save points/achievements on that application).

The benefits to the core data is simply that it's easier to implement and that it provides quick data retrieval, but the drawbacks would be the lack of scalability and the reliance on the phone itself in a world where phones break and get upgraded every month.

The benefits to internal API would be sheer amount of information that we could store, along with the ability to jump phones and even different platforms. However, it would be difficult to create and implement.

The benefits to GameCenter would be the ease of storing the data as well as allowing the user to compete with friends and followers on that application. However, this would require forcing the user to make an account, and would limit our data storage to what the application could take in.

After this analysis, Nupur and I decided to use the core data for V1, and potentially create an internal API for V2 to allow for scalability. While the GameCenter option is still attractive to us, we decided that we would push that as an optional feature in V2 as we didn't want to force the user to download another application.

External API Example

Please see the attached file! Out of our list of total features (see *List of Features: V1 & V2*), the features that require an external API are the following:

- The User can input a number of minutes and severity, and Outrun will create a route for their walk
- The User can input a distance and severity, and Outrun will create a route for their walk
- During the walk: Outrun will give user directions on where to go

In short, we need to be able to create a route with either a boundary on **duration** or **distance**, and provide directions throughout that route to the user. This requires two external APIs: Google Places and Google Maps, where the former is used to get a list of places and the latter is used to give us a route and directions between two places. Please see the example of both of those API calls in the playground.

Working Prototype & User Tests

Our prototype can be found here: https://www.figma.com/proto/L1AzHppJmJcYofhQtraZYQ/OutRun?node-id=1%3A2&scaling=scale-down.

The wireframes are almost the exact same from the previous sprint, with the addition of an example of seeing a zombie during the walk and how that interaction would play out.

Feedback from Users:

Feedback	Lessons Learned
User 1 ● Main user interaction makes sense and is intuitive ● "cute design" ● Interaction with the information screen is confusing ○ Would prefer cards when opening the app as a new user and then getting hints like a video game ○ Doesn't think a dedicated information section is necessary	We both assumed that we should make the app as easy to use as possible, but in doing so may have lost the ability to trust that our users understand the game. Instead of relying on text to show the user how to play, we can think of something new and easy.
User 2 ● Would like to see page with instructions first ○ Didn't notice the "i" in the corner, so needs it to be more visible ● App "makes logical sense" ● Zombie looked great! Enjoyed the tasks, would like to see even more ● Variety in zombie characters - didn't understand the "blood"	Definitely didn't realize that the application is in fact slightly complicated, as we've been working on it for so long. The feedback to make the instructions more salient is something we definitely plan on implementing. The "blood" user 2 was referring to was a "red glow" we were using to indicate that a creature or action was incoming. We will definitely adjust that in the future, to make it more obvious.
User 3 ● First splash screen doesn't really inform me what the game is about ● Create and history -> summary is a clear flow ● Wants some dynamic information for the first time playing to show how to "slash" ● Didn't realize how to play (didn't see information button) until they accidentally hit it	We've gotten some pretty clear feedback that the mechanics of the game itself is difficult to pick up on the fly, and that the information screen tucked away in the corner won't be enough. We'll be fixing this for our first version of the app, and user testing to make sure that the first experience for the user is a clear one.

User 4 • The blood splash seems like an interface glitch; make the graphic clearer • Likes the overall game idea and loves the minimap • Would like the design to be more “complex/sophisticated”. Use gradients and realism more.	This is the second users to mention the “blood”, so it’s definitely a concern and something we want to adjust. The comment about our design is interesting, and something worth looking more into.

Mentor Feedback Report

In our meeting with Ashu, we asked her for any feedback on the internal data storage question. Here is her feedback.

Feedback	Response
Ashu is team “core data” and she recommends internal api/game center for v2	We agreed!
We were discussing an iteration option for the routes, and Ashu suggested that we actually allow the user to pick best out of 5 rather than try to think of an algorithm ourselves.	We loved the idea, but pushed it to V2 because it felt like an extra feature.
Ashu asked if our information section would be hardcoded or saved in the plist.	We discussed the options, and we most likely will hard code it unless we find that the text might have to be updated consistently.

Ashu:

- Use haptics when something is coming in
- Have the directions up right away
- Getting there and back (uturn?)
    - Would want to pick something halfway

    - Point A, B, & C - 3 destinations
        - B is halfway point
        - A & C are starting point
- ARKit??
    - Most important!!
    - Worst case, if you don't have directions, you can still play this game
- How to connect