Milestone 4

IFT 402 Capstone Project

By: Amber Cole
Bryce Kortlever
Arthur 'Evan' Schlemmer

Table of Contents

1. CHARACTERISTICS OF THE MAGIC ITINERARY MAKER

When creating the Magic Itinerary Maker, we were motivated by the idea that not every visitor to the Disneyland Resort would be familiar with all the activities that parks had to offer; the parks could be considered intimidating they have so many things to do. Our goal was to make it easier for people to plan their trip to the parks (ahead or during) so that they can get the most out of their trip.

<u>There are four main characteristics that accomplish this goal:</u>

1. Intuitive design for ease of use.
2. Convenience of a mobile application, user is delivered results quickly.
3. Comprehensive information stored in a database for easy access.
4. Cross-platform capable, reaching a wider range of users.

Each of these characteristics addresses a portion of our mission statement:

"A guided experience in the hand of every adult visitor; taking the time and effort out of planning by providing a comprehensive and efficient means of experiencing the Disneyland Resort in one smartphone application."

1.1 INTUITIVE DESIGN

We want the Magic Itinerary Maker to be easy to use, and our user interface design can be summed up into three words. The screen layouts in our app are simple, inline layouts. The user controls are made large and clearly labeled so that there is no ambiguity as to what they do. Much of our app is focused around text, so there are few ways of organizing our pages without making them a mess to read, especially on small mobile devices.

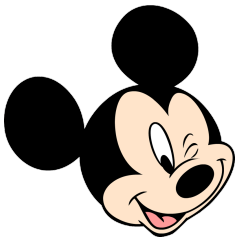
<u>There are a few simple rules we followed while designing the Magic Itinerary Maker:</u>

- User interaction will take place in the center of the screen, as opposed to the top or the bottom.
- Use as much of the screen as possible.
- Place navigation in standard areas (top left for back, bottom right for forward).
- Provide adequate spacing between elements.
- Order elements from most important at the top, downward.

1.2 CONVENIENCE

Mobile Applications, by nature, are convenient to use because they can be used anywhere. We wanted to take it a step further by making our app efficient so that users are delivered results quickly. We focused on efficiency on the user control side, as well as behind the scenes.

- *On the user side*, we kept user interaction to a minimum. There are no features or controls included that are not essential to the user experience. If there are fewer screens and fewer buttons between the user and their itinerary, they will get their itinerary faster. For example: the Custom Itinerary feature only prompts the user to respond for four questions, where originally we had over 10 possible questions planned.
- *On the backend,* we kept our code light. As development on the Magic Itinerary Maker progressed, we tried to keep our code efficient by avoiding processing large amounts of data of any kind (GUI or itinerary generation). Animated page transitions, button rendering, and itinerary generation are kept basic, as these features can easily bloat our code.



The use of a database has also made the Magic Itinerary Maker convenient for us, the developers, leading into the next section...

1.3 COMPREHENSIVE DATABASE

While simple, the Magic Itinerary Maker's database is the heart of the app. We wanted to provide users with all the information they would need to navigate the Disneyland Parks so we started with the largest activity type, ride attractions.

<u>Our database has all the information a visitor to the park would need about an attraction:</u>

- The name of the attraction.
- Which park the attraction it is located.
- The area inside the park the attraction is located.
- Any height requirements an attraction may have.
- Short description of the type of attraction.
- FastPass availability of each attraction.
- An image of the attraction for new visitors.
- The Magic Itinerary Maker priority rank.

The information above is mutually beneficial to both the user and, we, the developers; we can generate user itineraries as well as provide them with this information to the user for their own use. And as the app grows the database will easily do so too, making activity additions (such as show entertainment or dining) simpler in the long term.

1.4 CROSS-PLATFORM CAPABILITIES

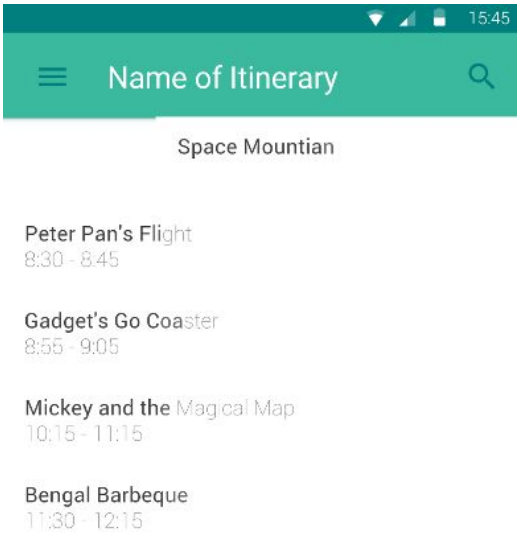
While we will be demonstrating it on Android devices, the Magic Itinerary Maker has the ability to be cross-platform without recreating the app in another programming language. Due to the nature of Cordova and it's use of native WebViews, the Magic Itinerary Maker can easily be ported to many platforms including iOS, Windows, Blackberry, in addition to our focus, Android.

As much as we would have liked to have deployed it across multiple platforms, an overall lack of resources (no devices to actually use for deployment) has kept this characteristic low on the priority list. Nevertheless, it is an important characteristic of our app, and one of the reasons we decided to develop using Cordova.

2. HOW WE GOT HERE

Decided on the idea of an Itinerary Maker.

"This project seeks to help enhance visitors' experience at the park by providing a personally tailored and efficiently streamlined itinerary generated from the users' timeframe, interests, and prior experience, with the ability to provide live-updates as certain events occur (ride-closures, increased wait-times, etc.), available as a smartphone app on the Android Platform"
-401 Write up



Began prototyping in Android studio.
Start realizing we might have bit off more than we can chew.
Decided not to use Android Studio.

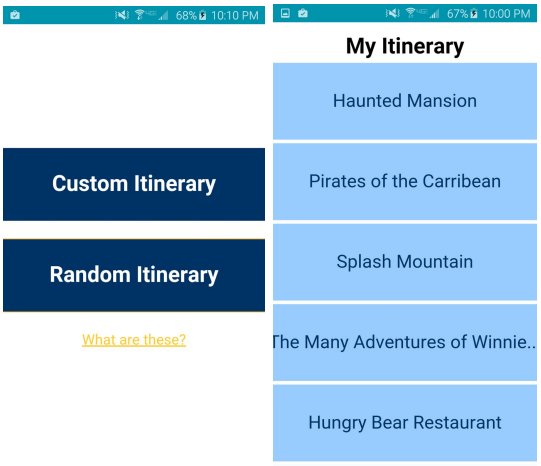
Switched to Cordova.
Reevaluted feature set.

"The plan to accomplish development of this project: playing to our strengths and focus."
-Milestone 1

"The app will be driven by two main features to achieve the project mission:

1. Generating Itineraries
2. Providing Comprehensive Information"

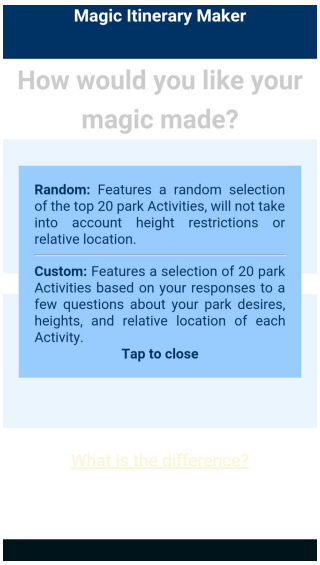
-Milestone 2

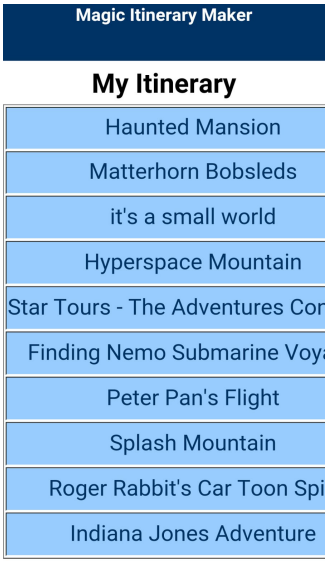


Started developing GUI elements.
Compiling of database information begins.

First version of database completed.
Work on GUI response begins.
Magic Itinerary Maker gets it's name!

The first button to ever work:
"What's the difference?"





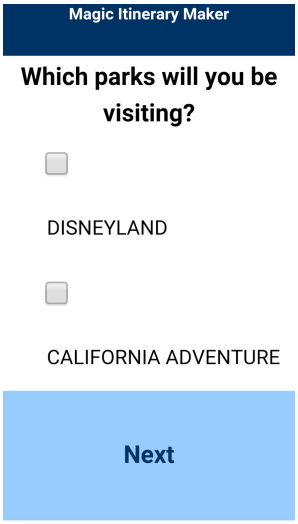
We can change pages!

Random Itinerary Generation development gets off to a rough start.

But it gets there by the weekend.

More buttons working too!

The app gets some love from GIMP and gets a splash screen and an icon.



The beast of the Custom Itinerary begins.
Database is altered to have more tables.

Struggles with strange rendering issues.

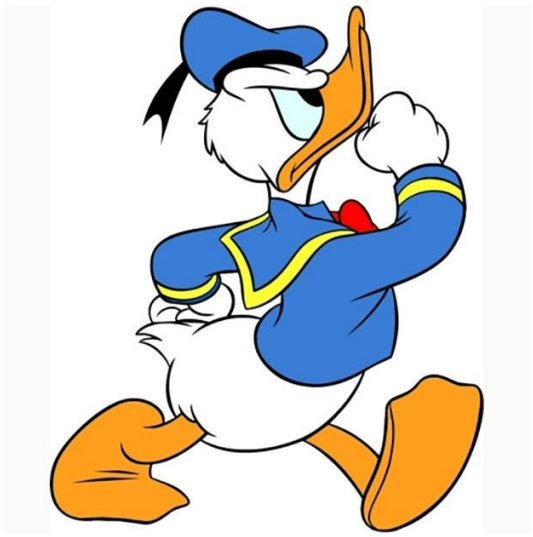
Custom Itinerary WORKS.
The app WORKS!

Design gets some love to include:
Page transitions
DISNEY EVERYWHERE
More colorful-ness

Final testing continues until showcase!

Donald Duck says:
"Why don't you have screenshots yet?"

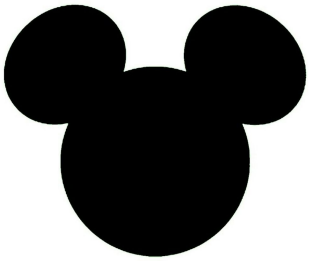
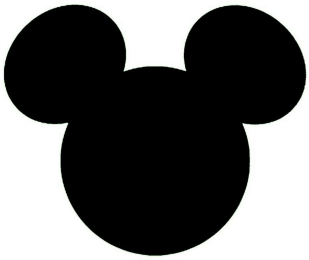
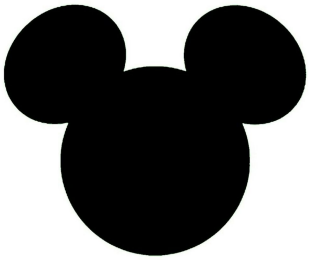
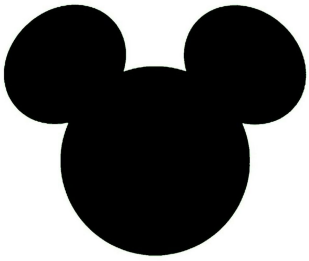
We will, Donald, we will.



FINISHED PRODUCT!

3. USER GUIDE

The Magic Itinerary App is easy to use!

#1 **Start by downloading the app onto your device of choice.** **Then simply tap the MIM icon to start!**	 Screenshots to come!
 Screenshots to come!	**#2** **Once the splash screen fades, you're ready to create your own itinerary for the Disney Parks.** **Simply tap on the type of Itinerary you would like to create.** **You can tap "What's the Difference?" to learn more about each type of itinerary!**
#3 **You will be prompted to answer some questions no matter which itinerary you choose.** **Just continue answering questions and tapping "Next".**	 Screenshots to come!
 Screenshots to come!	**#4** **Once you are finished with the questions your itinerary will appear!** **Tap on any of the attraction names to reveal more information about it.** **HAVE FUN AT THE PARK!**

Milestone 1

IFT 402 Capstone Project

By: Amber Cole
Bryce Kortlever
Arthur 'Evan' Schlemmer

1. OVERVIEW

Vision Statement: Elegant Crowd Control. A synergetic means of enhancing visitor experiences and maximizing park utilization. The end goal would be a system that could provide users with real time information about the parks, notifications pushed by Disney itself based on their resources.

Mission Statement: A guided experience in the hand of every adult visitor; taking the time and effort out of planning by providing the most comprehensive and efficient means of experiencing the Disneyland Resort in one smartphone application.

Target Audience: Smartphone users, primarily ages 18-49, visiting the park in the next calendar year, with 5 or fewer trips in the last decade to the Disneyland Resort in Anaheim CA (approximately 10 million people currently, park attendance$_1$ and smartphone usage$_2$ trending upwards). Will henceforth be referred to as "*inexperienced visitors*" or "*users*" in this document.

The motivation behind the creation of the app stems from being an experienced Disneyland visitor and discussing the park with inexperienced visitors who have been overwhelmed by the expansiveness of the parks. We believe that there is a method to planning a visit the parks that can optimize a visitors time, allowing for a more immersive experience without having to do extensive research to plan before or during the trip. The method: prioritizing and focusing on fewer activities. The project will focus on providing a mobile app that utilizes this method for these inexperienced visitors to use while planning and/or experiencing a trip to the parks. The future success of the project will rely upon a solid foundation of itinerary generation (instant and custom options) that provides the user with an efficient and unique route* around the parks, with more comprehensive features to come later.

Strategy:

5. Intuitive design for ease of use.
6. Convenience via mobile application.
7. Cross-platform, reaching a wider range of users.
8. Comprehensive information, no need for outside sources.
9. First to market, no other app of this kind exists.

2. CHARACTERISTICS

There are two main features of the app:

1. **Generate Itinerary** (First Sprints, Top Priority)
    1. Random Generation provides the user an itinerary immediately with limited input required after selection. This itinerary is comprised of a random selection from the top 20 rated activities within the database. Does not provide efficient routing* between activities.
    2. Custom Generation provides the user an itinerary after a series of short questions related to activity priorities of user and their park preferences. This itinerary is comprised of activities relevant to user responses. Provides an efficient route* between activities.
2. **Provide Information** (Final Sprints, Room for Expansion)
    1. Provide useful information about park activities easily accessible from the itinerary screen.
        1. Descriptions
        2. Map locations
        3. Height Requirements

Design: Quick and Simple. Keeping user input straightforward and to a minimum so that users can get the information about the parks they need quickly is going to be the primary draw of the app. If it is fast, easy, and provides information users want, they will continue to use the app$_3$.

3. SPRINT SCHEDULE & TASK BREAKDOWN

Note: All tasks and features following should be assumed high priority unless otherwise noted. Time has been made available during the final sprint to incorporate any low priority features deemed feasible or necessary.

Sprint 1: Feb. 8 - Feb. 22
1. Get Cordova running
2. Feature 1.1 control
3. Database evaluation

Sprint 2: Feb. 22 - March 7
1. Feature 1.1 complete
2. Feature 1.2 control
3. Database reevaluation

Sprint 3: March 7 - March 21
1. Feature 1.2 complete
2. Save option integrated
3. Begin Work on Feature 2

Sprint 4: March 21 - April 4
1. Feature 2 base requirements met
2. Finalize Database
3. Begin focus on GUI usability

Sprint 5: April 4 - April 25
1. Testing
2. Beautify
3. Extra time incase of delays

Product Debut Preparation: April 25 - May 2
1. Presentation Materials
2. Mock Questions & Rebuttal Preparation

Backlog	To Do
1. Design interface for Custom Itinerary questions 2. Properly save itineraries in local storage 3. Design interface for ride information 4. Correctly display ride information within generated itinerary 5. Dynamic information 6. Finalize Database 7. Focus on GUI	Sprint 2 1. Design screen for Random Generation 2. Design basic itinerary presentation after generation 3. Develop basic algorithm for generating custom itinerary
In Progress	**Completed**
Sprint 1 ends: **02/22** **Amber** 1. Complete formatting of raw database info 2. Help with JavaScript development **Bryce** 1. Complete Database table set and import data 2. Begin work on Shake Things Up feature a. Top 20 DB b. Adapt Java code for random itinerary **Evan** 1. Work with Bryce on Shake Things Up feature a. Explain java code that is being adapted for this use. **TEAM** 1. GET CORDOVA WORKING	**Amber** • Prototype • Compiled database info • Develop method of rating important attractions **Bryce** • Database Modeling **Evan** • Start developing in Android studio o Logo o Splash screen o Question entry • Milestone 1 Docs **Team Effort** • Question Development

4. THE PLAN

The plan to accomplish development of this project: playing to our strengths and focus. Given the change to development on the Cordova framework, all team members have experience with using the given languages (HTML, CSS, Javascript) enabling more focus on development. With only two main features of the app, we have narrowed our area of focus allowing plenty of time to successfully complete those features and hopefully add features originally proposed in 401 without overwhelming the team with "feature creep".

Team Roles & Responsibility
- Amber Cole
  - Product Owner
  - Designer (Primary)
  - Documents Specialist
  - Developer
- Arthur Schlemmer
  - Developer
- Bryce Kortlever
  - Lead Developer (Primary)
  - Database Specialist

The virtual backlog featured in part 3 of this document, hosted via google docs, serves as the central source for project, feature, and task status. As tasks from the backlog are chosen or delegated, the relevant team member completes the task and is responsible for submitting it to the PO for final approval and signoff at which time she will clear that team member to adjust the backlog accordingly. The Project Owner retains the sole authority to delegate tasks to team members however, in keeping with the spirit of Agile methodology team members will for the most part be responsible for choosing their own contributions based on their official roles and individual strengths.

Meetings: Regular, weekly team meetings will be held on Tuesdays and Thursdays from 2:45 - 4:30pm.

Mentor Involvement: The project's mentor has made themselves available via email and virtual collaboration for testing, debugging, and general advisement. There will be weekly progress reports sent on Fridays by the PO, as well as Milestone reviews done a few days before submission dates.

Milestone 2

IFT 402 Capstone Project

By: Amber Cole
Bryce Kortlever
Arthur 'Evan' Schlemmer

1. CONCEPT

Mission Reminder: Taking the time and effort out of planning by providing a comprehensive and efficient means of experiencing the Disneyland Resort in one smartphone application.

The app will be driven by two main features to achieve the project mission:

1. **Generating Itineraries**
2. **Providing Comprehensive Information**

1.1 GENERATING ITINERARIES

Random Generation: Random generation is for the visitor to the parks that wants immediate results, generally visitors who do not have a plan in place for their Disneyland visit. We want to provide a fast and easy way of providing them an itinerary, with little input required on the user's end.

Strategy: If chosen at the *main selection screen*, will provide the user an itinerary and move to the *itinerary screen* immediately. This itinerary is comprised of a random selection from the top 20 rated activities within the database, driven by javascript, and will not take into account relative locations of activities. Users will be able to move from the itinerary screen to the *activity description screen* and the **map location screen*.

Custom Generation: Custom Generation is for the visitor who has time to plan, whether it be the night before a visit to the parks or several months before. The custom option will provide these users with itineraries that align with their specific activities priorities taken into account, requiring more input by the user initially.

Strategy: If chosen at the *main selection screen*, will prompt user to respond to questions provided on a series of *question screens* followed by the *itinerary screen* when completed. These questions will be related to activity priorities of user and their park preferences. This itinerary is comprised of activities relevant to user responses and organized, using javascript, based on a popularity ranking of the activities within the database. This option will take into account relative locations of activities chosen for the itinerary. Users will be able to move from the itinerary screen to the *activity description screen* and the **map location screen*.

1.2 COMPREHENSIVE INFORMATION

This feature is to provide useful information about park activities easily accessible from the *itinerary screen*.

- Short descriptions of activities (ex. Dark, Thrill, Seating Available...) including height requirements. Located on *activity description screen*.
- Activity locations within parks *relative to user location. Located on *map location screen*.

Comprehensive information about the parks is essential to achieving the project mission, it makes navigating the parks easier. By providing information about park activities within our app, users can make decisions about these activities to fit their desires and needs, no matter which itinerary generation they choose. An example would be a family with children under 40 inches tall can see height requirements within our app and decide to skip any rides that have those requirements before they even enter the park.

2. WHY CORDOVA?

VS

ANDROID

APACHE CORDOVA™

"Android provides a rich application framework that allows you to build innovative apps and games for mobile devices in a Java language environment."[1]

Strengths:

- Speed and efficiency
- Access to Hardware

"Apache Cordova is an open-source mobile development framework. It allows you to use standard web technologies such as HTML5, CSS3, and JavaScript for cross-platform development." [2]

Strengths:

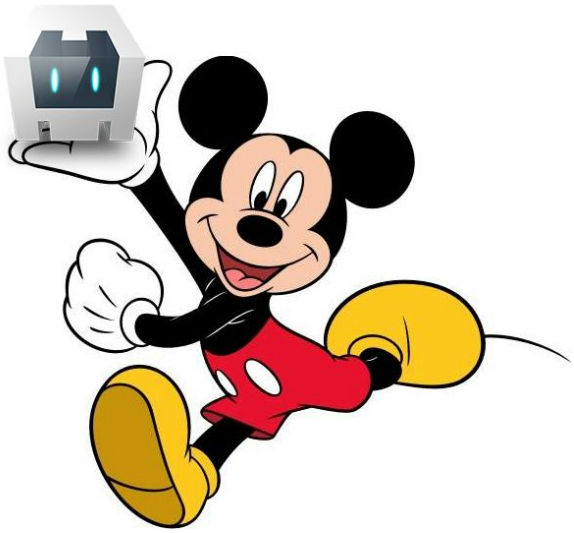
- Portable across most phone platforms
- Faster development

Of the choices presented, Android is the more powerful of the two frameworks providing easier access and efficient use of device hardware. However, when we consider what our app needs from a framework, the focus comes to portability and ease of development. We have no need for intensive use of a device's hardware to accomplish itinerary generation and presentation, we need it to be simple to develop and easy to use, neither of which depend on native app development. Cordova allows for cross-platform development without having to develop separate applications to be cross-platform as we would have to using Android (and any other native framework).

Cordova is what Team Disneyland needs.

As stated in Milestone 1, we as a team are playing to our strengths. We all have experience in Cordova's mediums (HTML, CSS, Javascript), enabling us to develop the app in a format that we are comfortable with. We will be able to accomplish both quality and quantity when it comes to project features as well as reach a wider audience with less effort and time invested with Cordova.

While Android was our first option, we realized once we started developing that is was going to be more difficult to achieve our project mission than we had previously anticipated. The switch to Cordova brings confidence in our ability to accomplish our project mission.

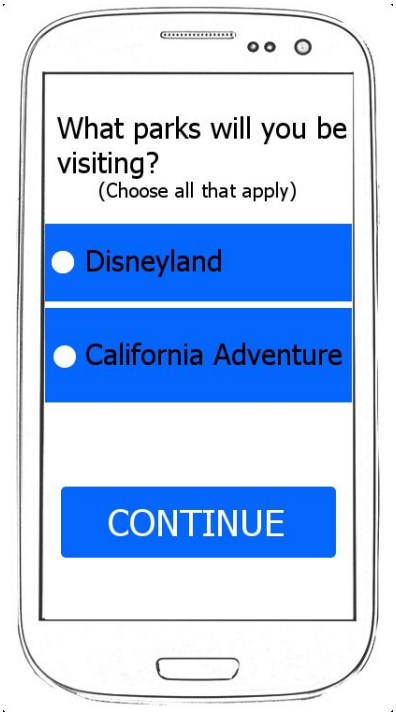


3. DEMONSTRATION

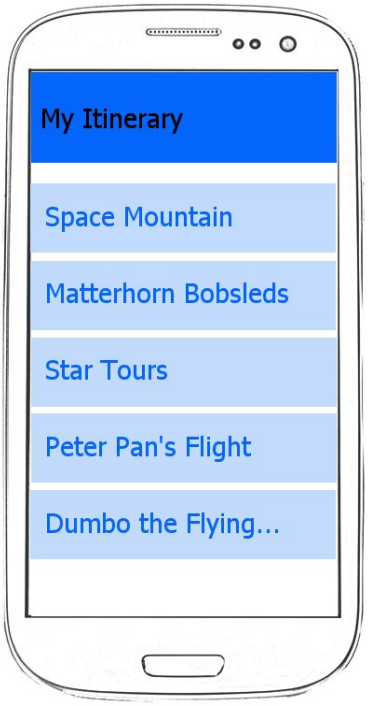
As per the Cordova framework, all screens will be formatted in HTML and styled using CSS.

Main Selection

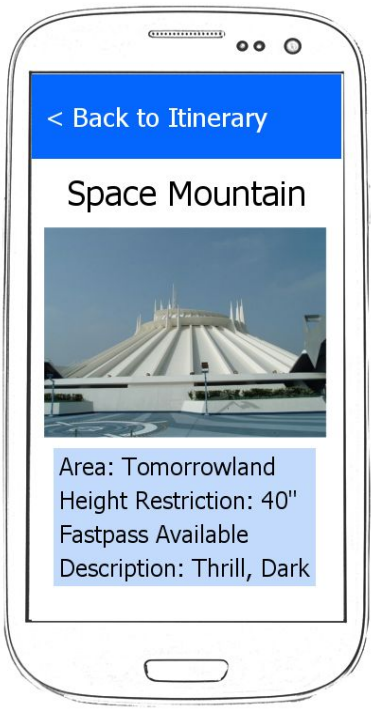
Sample Question



Itinerary



Activity Description



4. SCHEDULE UPDATE

Sprint 2: Feb. 22 - March 7
1. Random Generation feature complete, including functional GUI.
2. Custom generation control javascript working as designed.
3. Database reevaluation (is it up to date?).

Sprint 3: March 7 - March 21
1. Custom Generation feature complete, including functional GUI.
2. Save option integrated into app.
3. Begin Work on Feature 2 (Providing information: Activity Descriptions).

Sprint 4: March 21 - April 4
1. Feature 2 base requirements met (does the Activity Description display properly?).
2. Finalize Database
3. Begin focus on GUI usability, unifying style across all app screens.

Sprint 5: April 4 - April 25
1. Testing of all features.
2. Beautify design.
3. Extra time incase of delays.

Product Debut Preparation: April 25 - May 2
1. Presentation Materials
2. Mock Questions & Rebuttal Preparation

PREVIOUS SPRINT: Feb. 8 - Feb. 22
1. Setup Cordova framework (all team members)
2. Random Generation control javascript working as designed.
3. Database evaluation (is it comprehensive enough?)

Features list:
1. Generate Itinerary
    1. Random Generation
    2. Custom Generation
2. Provide Information

5. REFERENCES

1. Android description and images: https://developer.android.com/guide/index.html
2. Cordova description and images: https://cordova.apache.org/docs/en/latest/guide/overview/
3. Phone Stock Image: http://tuimages.photodeck.com/
4. Disneyland Map: http://disney.wikia.com/wiki/Disneyland_Maps_Gallery

Milestone 1

IFT 402 Capstone Project

By: Amber Cole
Bryce Kortlever
Arthur 'Evan' Schlemmer

1.1 SCREEN LAYOUTS

Screen layouts in our app will be simple, inline layouts. Much of our app is focused around text, so there are few ways of organizing our pages without making them a mess to read, especially on small mobile devices.

There are a few simple rules we follow when designing our screen layouts:

1. User interaction will take place in the center of the screen, as opposed to the top or the bottom.
2. Use as much of the screen as possible.
3. Place navigation in standard areas (top left for back, bottom right for forward).
4. Provide adequate spacing between elements.
5. Order elements from most important at the top, downward.

1.2 USER CONTROLS

There are three rules we follow when designing user controls:

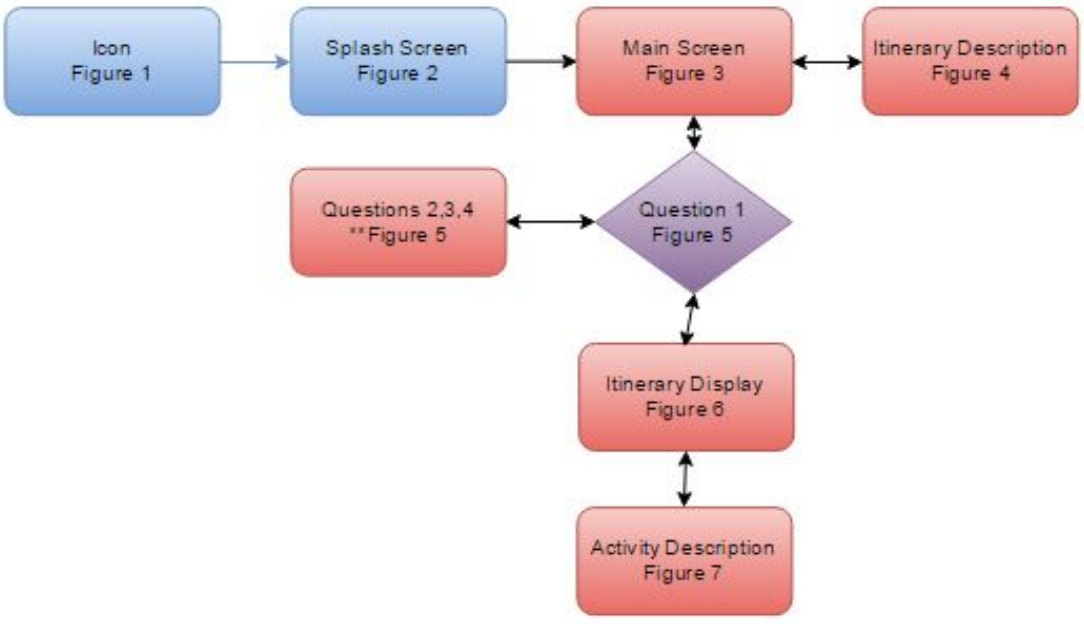
1. Make it large.
2. Make it obvious.
3. Make it scalable.

1.4 DATABASE USAGE

The database is called upon to fill in certain information on *Figures 6 and 7*. Figure 6, the Itinerary Display, has the database filling in the Activity titles; Figure 7, the Activity Display, has the database filling in all information about the Activity, including the images.

1.4 SCREEN FLOW

From the beginning, the flow of our app has felt intuitive. Building and presenting an itinerary is similar to taking an online survey: ask some questions, then display the results. We've tried to keep the flow of our app simple, keeping the number of different screens to a minimum (5 *unique* screens total).



**The primary font used in this app is Tahoma, unless otherwise noted.

icon.png

Launch icon for app

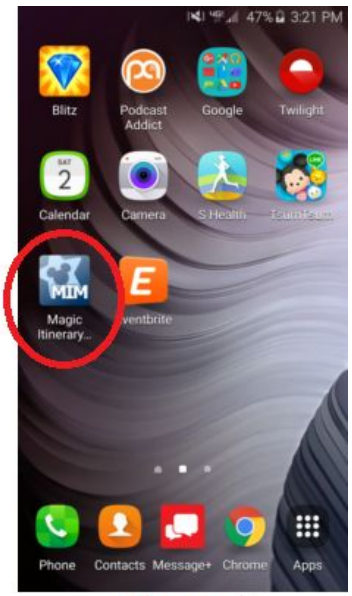


Figure 1

splash.png

Splash Screen stays on screen for 5 seconds

Figure 2

Main Screen

User interaction starts here

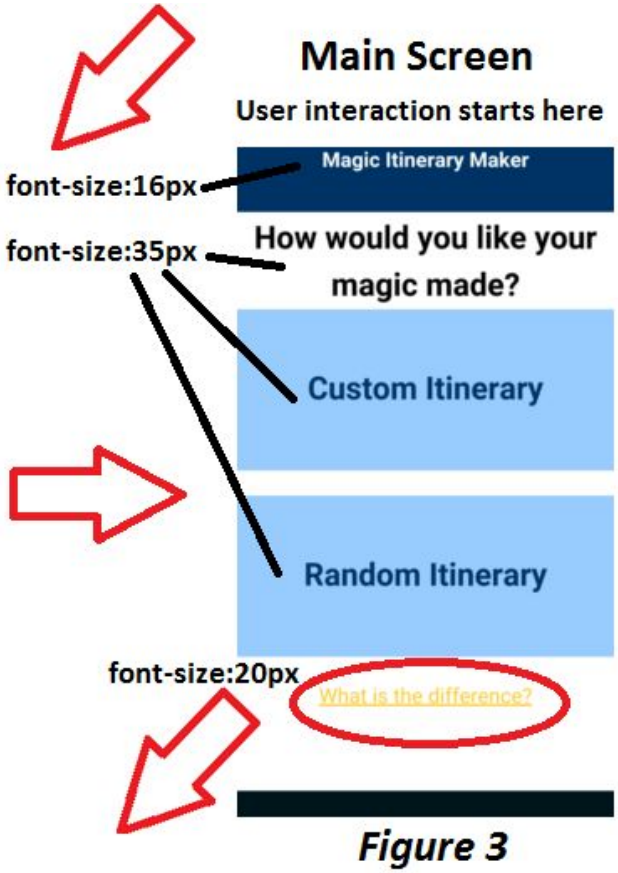


Figure 3

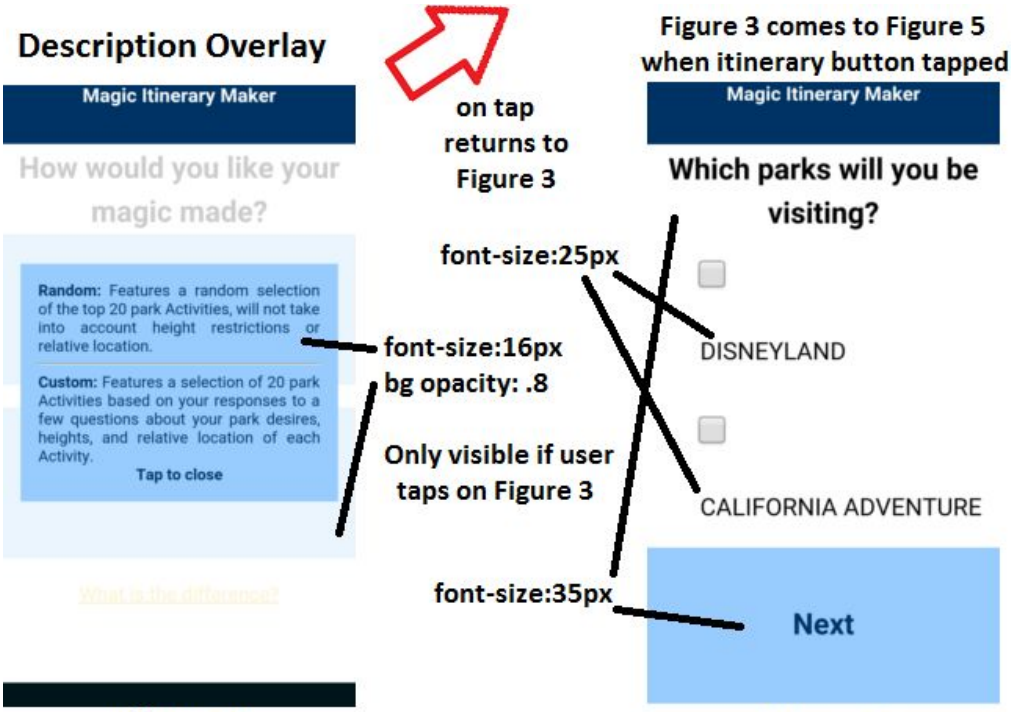


Figure 4

Figure 5

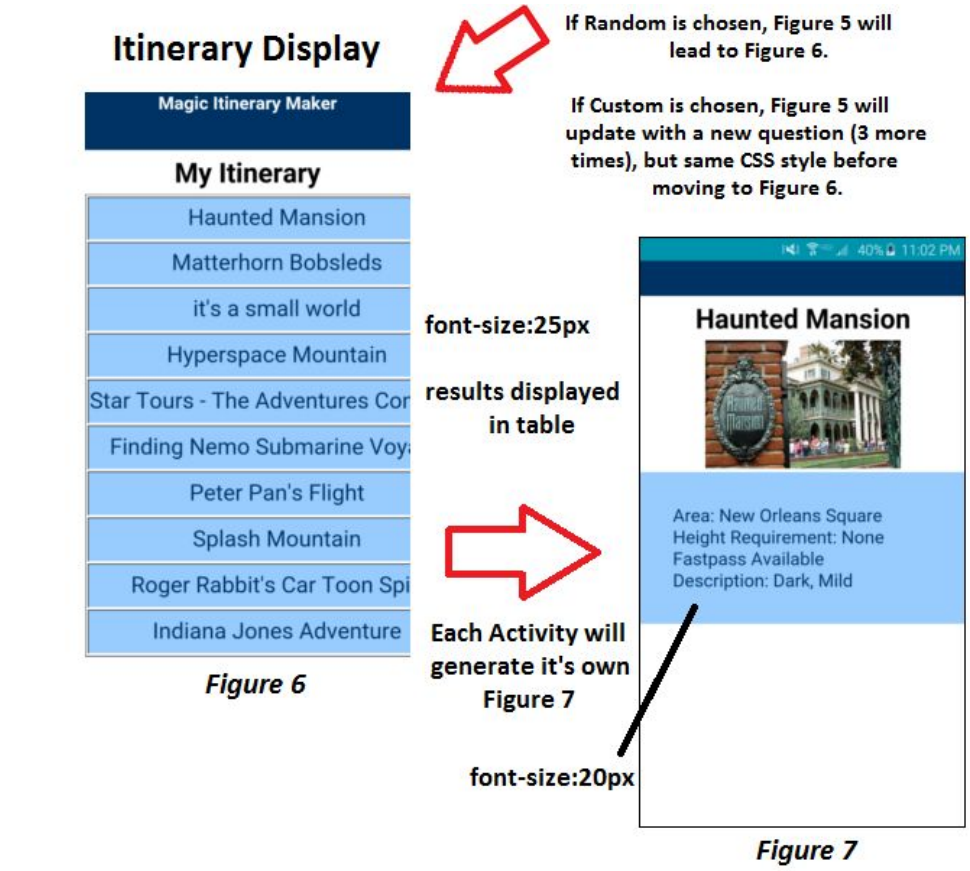


Figure 6

Figure 7