

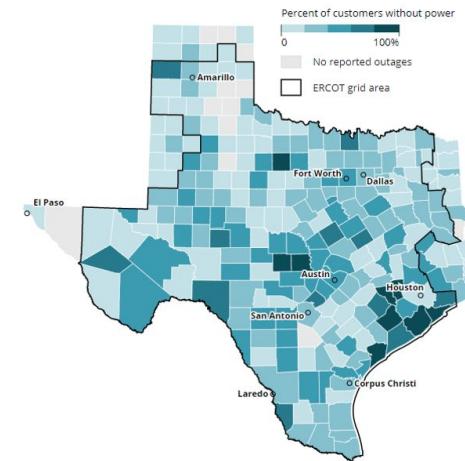


Team 28: Power Outage Education App Bi-Weekly Update 2

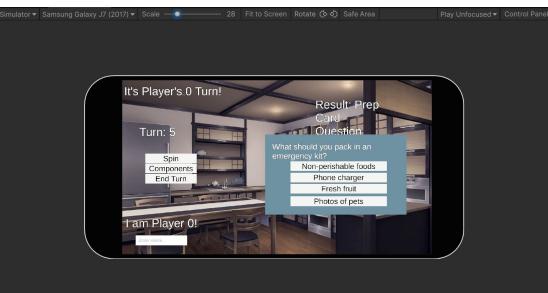
Jackie Villanueva, Aidan Petropoulos, Joey Raphael
Sponsor: Dr. Mladen Kezunovic
TA: Swarnabha Roy

Project Summary

- Problem statement: “People are often uninformed about the best course of actions to take before and during a power outage. This lack of knowledge can result in lack of preparation and uninformed decision-making that can cause further harm.”
 - The 2021 Texas Freeze
 - Outages caused by natural disasters
- Solution: Develop an app that provides different age-specific knowledge to educate students about the impact of power outages utilizing engaging, interactive apps. By informing people about what measures to take before, during, and after a power outage, the harm caused by power outages can be mitigated.



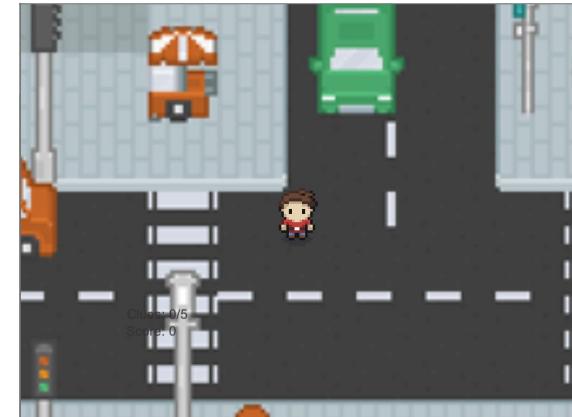
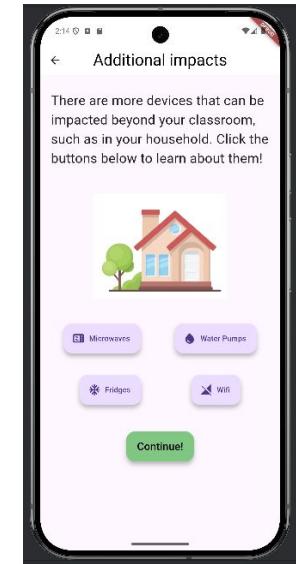
Project/Subsystem Overview



Power Outage Lesson
 (Flutter | Jackie)

Board Game
 (Unity | Joey)

LittleDoers Game
 (Unity | Aidan)



Project Timeline

Base Subsystem Functionality (completed 9/14)	Flutter/Unity Integration (to complete 9/28)	Refine Subsystems (to complete by 10/25)	Testing with Students (to complete by 11/15)	Finalize App Design (to complete by 11/29)	Demo (to complete by 12/4)
---	--	--	--	--	----------------------------

- Green: Completed
- Yellow: In Progress
- Red: Behind Schedule
- White: Not started

Elementary Subsystem

Aidan Petropoulos

Accomplishments since last update 20 hrs of effort	Ongoing progress/problems and plans until the next presentation
<ul style="list-style-type: none">- Created temporary Flutter project to isolate and debug Unity plugin issues (<code>flutter_unity_widget</code>)	<ul style="list-style-type: none">- Resolving Gradle build issues (namespace errors, Kotlin version mismatch) during Unity–Flutter linking- Integrating <code>flutter_unity_widget</code> plugin into main Flutter app for embedding Unity content- Verifying Flutter to Unity communication channel

Elementary Subsystem

Aidan Petropoulos

```
> Configure project :webview_flutter_android
WARNING: Using flatDir should be avoided because it doesn't
         work correctly with the new build system.
         It is recommended to use the dependency resolution
         mechanism built into the new build system.
         See https://gradle.org/gradle-6.0-deprecation-guide.html#flat-dir-for-gradle-build-systems
         for more on this, please refer to https://docs.gradle.org/6.0/javadoc/org/gradle/api/artifacts/configurations/Configurations.html#deprecations-in-gradle-6.0
         Deprecated Gradle features were used in this build, making it incompatible with
         future versions of Gradle. It is recommended to update your build to use the
         latest features.

         You can use '--warning-mode all' to show the individual deprecations
         or warnings for each feature.

For more on this, please refer to https://docs.gradle.org/6.0/javadoc/org/gradle/api/artifacts/configurations/Configurations.html#deprecations-in-gradle-6.0
         deprecations in the Gradle documentation.

BUILD SUCCESSFUL in 23s
13 actionable tasks: 4 executed, 9 up-to-date
```



K-5 Elementary



6-8th Middle School



9-12th High School

Middle School Board Game

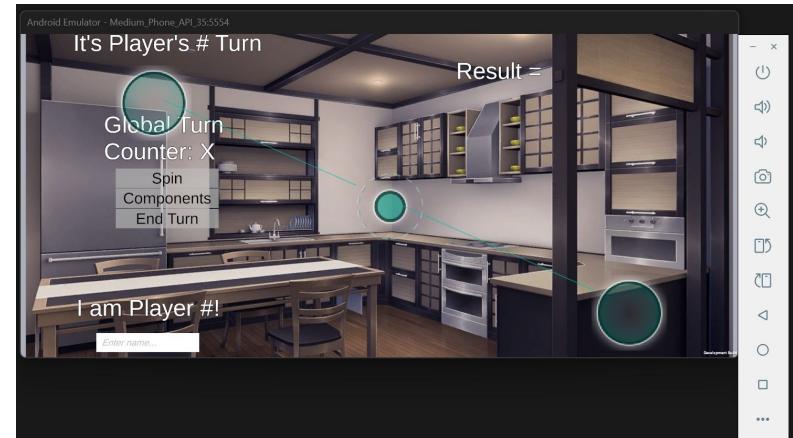
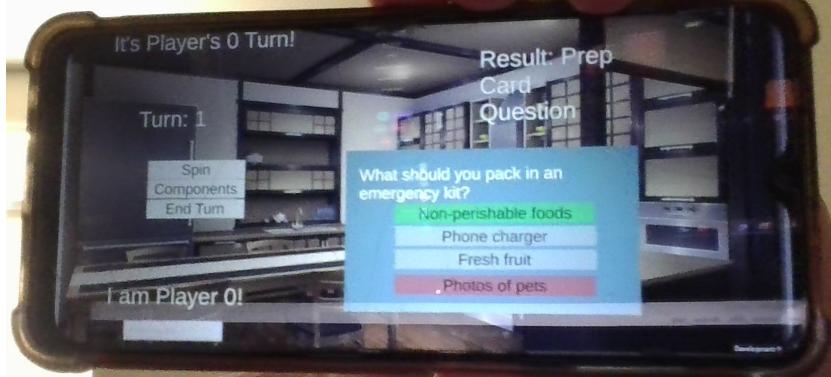
Joey Raphael

Accomplishments since last update 20 hrs of effort	Ongoing progress/problems and plans until the next presentation
<ul style="list-style-type: none">- Optimized memory / frame performance	<ul style="list-style-type: none">- Implement Flutter / Unity integration by embedding into Flutter web app- Verify that all UI elements on WebGL port are interactable- Identify codebase mismatches that prevent compatibility (Gradle, Unity)

Middle School Board Game

Joey Raphael

- Able to run game on mobile device, buttons are functional, behavior is expected
- Difficulty with Unity game embedding into blank Flutter app
 - Crashing, UI not scaling, response latency



High School Subsystem

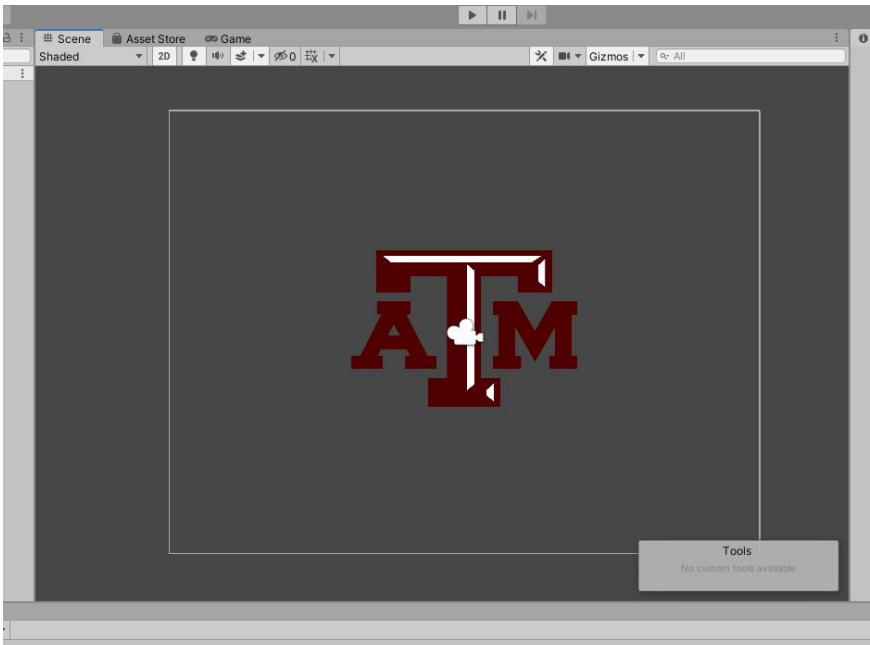
Jackie Villanueva

Accomplishments since last update 11 hrs of effort	Ongoing progress/problems and plans until the next presentation
<ul style="list-style-type: none">- Finalized integration method for Flutter/Unity- Created a temporary Unity project in different versions	<ul style="list-style-type: none">- Complete integration of a Unity project/Flutter on the different versions- Remove placeholders and refining UI for Flutter App

High School Subsystem

Jackie Villanueva

- Temporary project made on Unity (in Unity versions 6000.0.39f1 and 2019.3.6f1)
- Flutter base app prepared for integration

A screenshot of a code editor showing a Flutter project structure and a main.dart file. The project tree on the left includes ".dart_tool", ".idea", "android", "assets", "ios", "lib" (containing "high_school_page.dart" and "main.dart"), "linux", "macos", "unity" (containing "TemporaryProject" which itself contains "Assets", "Builds", "Library", "Logs", and "Packages"). The main.dart file on the right contains the following code:

```
//import 'package:english_words/english_words.dart';
import 'package:flutter/material.dart';
//import 'package:provider/provider.dart';
import 'package:flutter_unity_widget/flutter_unity_widget.dart';
import 'high_school_page.dart';

void main() => runApp(MaterialApp(
  home: Home()
)); // MaterialApp
class Home extends StatelessWidget { //Base page design, Home but
  const Home({super.key});
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Power Outage Education App', style: TextStyle(
          centerTitle: true,
          backgroundColor: Colors.blue,
        ), // AppBar
      ),
    );
  }
}
```

The line "import 'package:flutter_unity_widget/flutter_unity_widget.dart';" is highlighted with a red underline.

Execution Plan

	8/24/25	8/31/25	9/7/25	9/14/25	9/21/25	9/28/25	10/5/25	10/12/25	10/19/25	10/26/25	11/2/25	11/9/25	11/16/25	11/23/25	11/30/25	12/7/25	Date of Milestone	Owner of Milestone
Debug and Test Software																	11/22/25	Team
Research Flutter/Unity Integration																	9/6/25	Team
Fix any remaining subsystem bugs																	9/20/25	Team
Integrate Unity onto Flutter																	10/11/25	Aidan & Jackie
Procure tablet for testing																	9/27/25	Team
Testing on tablet																	11/29/25	Team
Verify proper data transfer / communication between Unity / Flutter																	11/1/25	Team
Optimize performance of Unity app running within Flutter app																	11/1/25	Joey
Verify UI scaling on different aspect ratios, resolutions																	11/1/25	Joey
Validate question design, record learning progress of students																	11/14/25	Joey
Add nonhorizontal navigation																	11/1/25	Jackie
Remove placeholders																	11/1/25	Jackie
Refine Flutter App UI																	11/1/25	Jackie
Add start screen for unity game																	11/1/25	Aidan
Finalize questions with Doseum																	11/1/25	Aidan
Refine DoSeum Game UI																	11/1/25	Aidan
Testing with Students																	11/1/25	Team
Finalize App Design																	11/29/25	Team
Final Demo																	12/4/25	Team

Validation Plan

A Milestone #	B Test	C Detail	D Validation Method	E Data	F Status
1 App Functionality Testing		App successfully opens and does not crash, lag, or glitch.	Boot up the app and navigate through all systems of the app.	N/A	Untested
2 Subsystem Objective Completion		Each subsystem can be played from start to end.	Run through each individual subsystem and reach the 'end.'	N/A	Untested
3 User Interface Consistency		The UI is functional and consistent when run on an Android phone or tablet.	Run the app on an Android phone and tablet. Confirm the correct functionality of each button.	N/A	Untested
4 Offline and Online Functionality		The app has different interfaces for if it is online or offline.	Run the software with and without wifi. Check if the interface correctly differs between the two scenarios.	N/A	Untested
5a Performance (Middle School System)		The game runs at an acceptable performance - Less than 100 ms latency between touch input and button response - Average of 30 FPS, frametimes less than 200 ms	Utilize profiling tools provided in Android Studio / Unity to record and track performance metrics	N/A	Untested
5b Multiplayer functionality (Middle School System)		User is able to communicate to another user in-game.	Simulate two instances of the app on the same 'network.' The two 'users' play and complete an instance of the game.	N/A	Untested
6 Nonlinear progression (High School Subsystem)		The user can navigate back to previous tabs that have already been accessed	Progress through the high school subsystem. Using the appbar on the top, confirm that previous 'tabs' may be accessed, then return to the most current tab.	N/A	Untested
7 Flutter and Unity		Unity apps successfully open and run without issue from Flutter.	Click Elementary and middle school tabs from Flutter base. Play through the individual games for both.	N/A	Untested
8 Learning Outcomes (Elementary)		Students demonstrate improved understanding of foundational concepts after playing.	Pre- and post-game quizzes, teacher feedback, observation.	N/A	Untested
9 Learning Outcomes (Middle School)		Students demonstrate improved understanding of targeted concepts after playing.	Pre- and post-game quizzes, teacher feedback, observation.	N/A	Untested
10 Learning Outcomes (High School)		Students demonstrate improved understanding of advanced concepts after advancing through the curriculum.	Post content quiz, teacher feedback, observation	N/A	Untested

Thank you for your time!

Any questions?