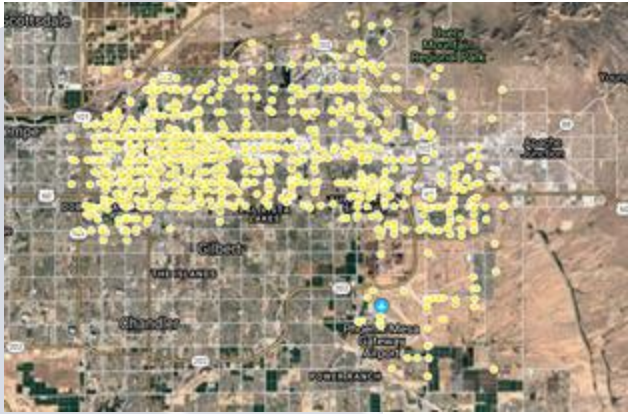
Transportation and Parking in a “Smart City”

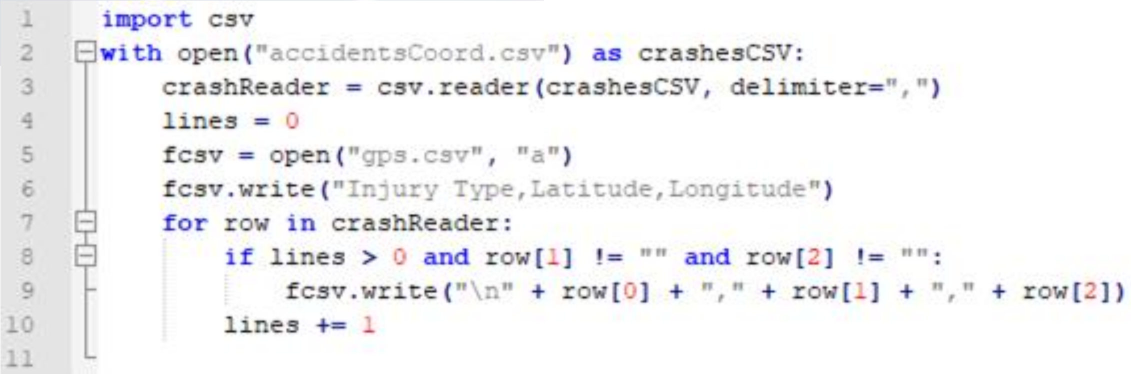
By: the Biotech Bunch

Parking Problem

- ASU Mesa Campus
  - Two less parking facilities
    - "Displaced 200 City employees" (1)
  - New students, faculty & staff needing parking
    - Projected min. ~2000 (2)
- Waymo
  - Need parking & charging station
- Property Damage & Safety Concern

Accident Analysis



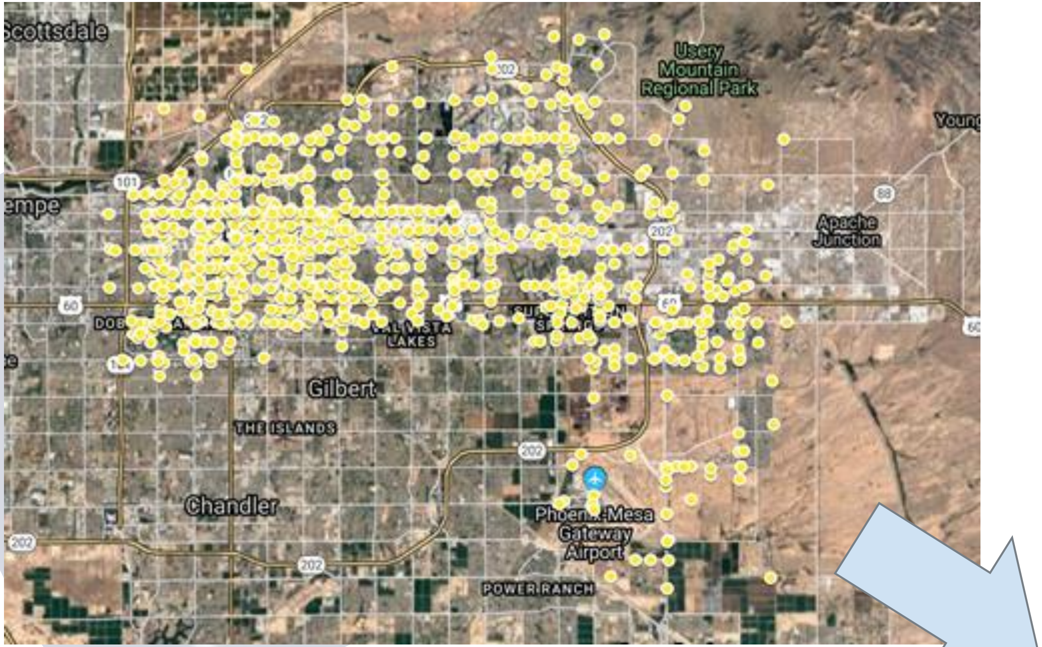


```python
import csv
with open("accidentsCoord.csv") as crashesCSV:
    crashReader = csv.reader(crashesCSV, delimiter=",")
    lines = 0
    fcsv = open("gps.csv", "a")
    fcsv.write("Injury Type,Latitude,Longitude")
    for row in crashReader:
        if lines > 0 and row[1] != "" and row[2] != "":
            fcsv.write("\n" + row[0] + "," + row[1] + "," + row[2])
        lines += 1

```

- City of Mesa Data Portal
- City of Mesa “Reported Traffic Accidents” data set
- All values without GPS coordinates were thrown out
- Remaining values were formatted to be inserted into Google Maps

Accident Analysis



- City of Mesa Data Portal
- Randomly selected 5 blocks to analyze accident location
- Found that 46% of reported accidents occur in parking facilities

Safety Consideration

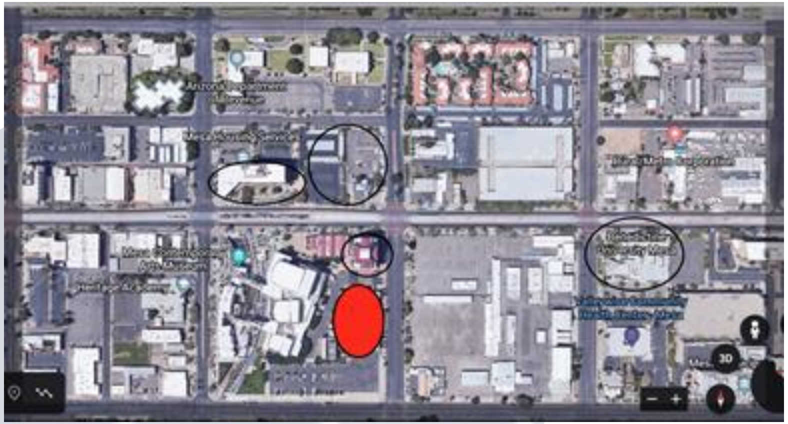
According to the National Safety Council, “more than 50,000 crashes occur every year in parking lots and garages in the U.S. These crashes result in approximately 500 deaths and 60,000 injuries every year” (3)

The Solution

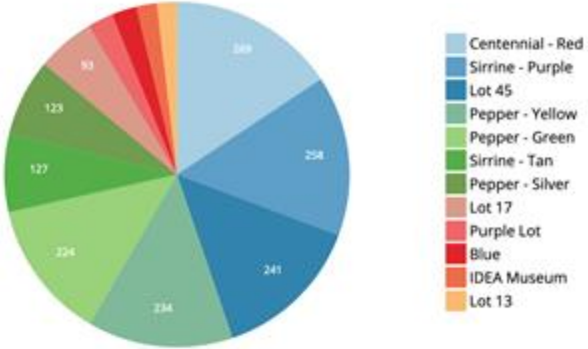
Innovative Parking Towers

- Maximizes Limited Space
  - Compact stacking & circular arrangement
    - 20 vehicles per story (space 9’ x 18’) with 10 stories =200 vehicles (115’ x 115’)
    - Parking lot holds 60
- Automated Lift
  - Reduced Accidents
  - Increased safety
- Green Potential
  - Solar Panels
  - Living Structure

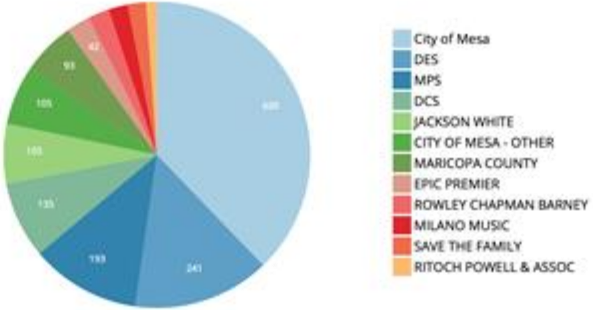
Location Determination



Locations with most parking permits held



Organizations with most parking permits

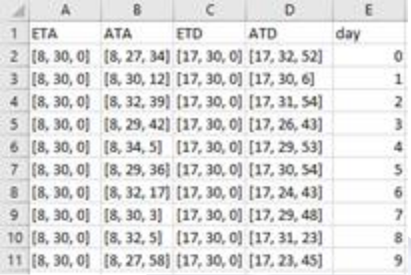


- Data obtained and rendered using Mesa Arizona’s database
- “Downtown Parking Permits” dataset

Power Consumption

- **Maximum Required Mechanical Power:** 26 kW (5,6)
- **Input Electrical Power:**
  - City Power: 35 kW (35%, 9 kW Lost as heat)
  - Solar: 32 kW (23.5%, 6 kW Lost as heat)
    - 182 square meters of Solar Panels needed (Power density of 175 W/m²)
    - $76,000 in solar panels

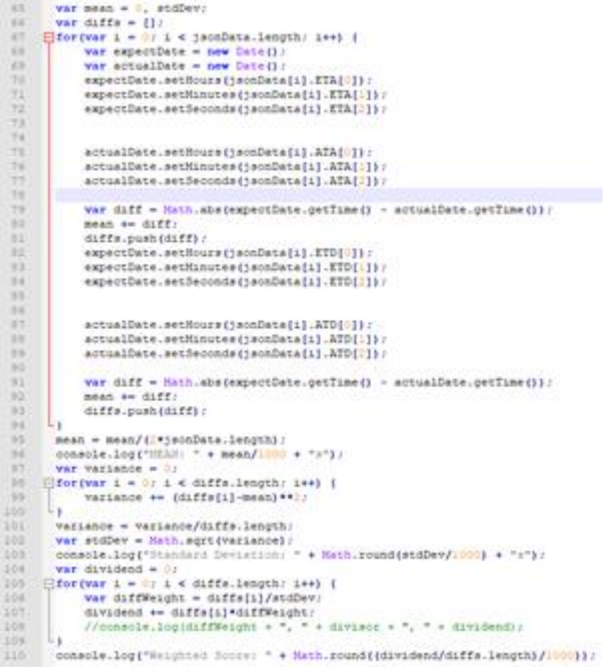
Parking App/Score



ETA	ATA	ETD	ATD	day
[8, 30, 0]	[8, 27, 34]	[17, 30, 0]	[17, 32, 52]	0
[8, 30, 0]	[8, 30, 12]	[17, 30, 0]	[17, 30, 6]	1
[8, 30, 0]	[8, 32, 39]	[17, 30, 0]	[17, 31, 54]	2
[8, 30, 0]	[8, 29, 42]	[17, 30, 0]	[17, 26, 43]	3
[8, 30, 0]	[8, 34, 5]	[17, 30, 0]	[17, 29, 53]	4
[8, 30, 0]	[8, 29, 36]	[17, 30, 0]	[17, 30, 54]	5
[8, 30, 0]	[8, 32, 17]	[17, 30, 0]	[17, 24, 43]	6
[8, 30, 0]	[8, 30, 3]	[17, 30, 0]	[17, 29, 48]	7
[8, 30, 0]	[8, 32, 5]	[17, 30, 0]	[17, 31, 23]	8
[8, 30, 0]	[8, 27, 58]	[17, 30, 0]	[17, 23, 45]	9

Sample data for testing

JavaScript code for score calculation



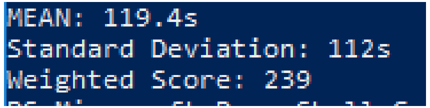
```
var mean = 0, stdDev;
var diffs = [];
for(var i = 0; i < jsonData.length; i++) {
    var expectDate = new Date();
    var actualDate = new Date();
    expectDate.setHours(jsonData[i].ETA[0]);
    expectDate.setMinutes(jsonData[i].ETA[1]);
    expectDate.setSeconds(jsonData[i].ETA[2]);


    actualDate.setHours(jsonData[i].ATA[0]);
    actualDate.setMinutes(jsonData[i].ATA[1]);
    actualDate.setSeconds(jsonData[i].ATA[2]);

    var diff = Math.abs(expectDate.getTime() - actualDate.getTime());
    mean += diff;
    diffs.push(diff);
    expectDate.setHours(jsonData[i].ETD[0]);
    expectDate.setMinutes(jsonData[i].ETD[1]);
    expectDate.setSeconds(jsonData[i].ETD[2]);


    actualDate.setHours(jsonData[i].ATD[0]);
    actualDate.setMinutes(jsonData[i].ATD[1]);
    actualDate.setSeconds(jsonData[i].ATD[2]);

    var diff = Math.abs(expectDate.getTime() - actualDate.getTime());
    mean += diff;
    diffs.push(diff);
}
mean = mean/(2*jsonData.length);
console.log("MEAN: " + mean/1000 + "s");
var variance = 0;
for(var i = 0; i < diffs.length; i++) {
    variance += (diffs[i]-mean)**2;
}
variance = variance/diffs.length;
var stdDev = Math.sqrt(variance);
console.log("Standard Deviation: " + Math.round(stdDev/1000) + "s");
var dividend = 0;
for(var i = 0; i < diffs.length; i++) {
    var diffWeight = diffs[i]/stdDev;
    dividend += diffs[i]*diffWeight;
    //console.log(diffWeight + ", " + divisor + ", " + dividend);
}
console.log("Weighted Score: " + Math.round((dividend/diffs.length)/1000));
```



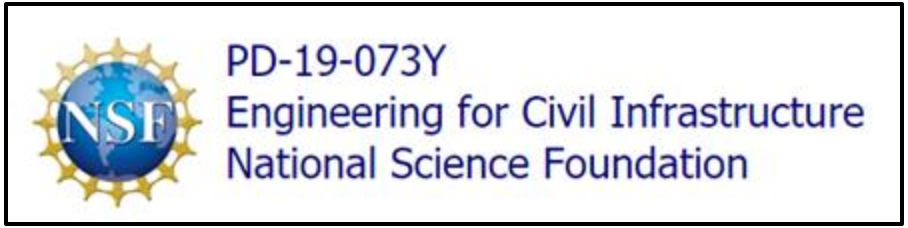
```
MEAN: 119.4s
Standard Deviation: 112s
Weighted Score: 239
```

Score Calculation Output

- Each customer has “reliability score”
  - Score goes down when customer is on time
  - Score increases when customer is later/earlier than expected
  - Customers with lower scores have priority
- Simple score calculation code was written and tested with dummy data
- App can integrate with Waymo’s ride hailing service

Cost

- Carvana vending machine averages $5 million
- ASU Tempe charges $210-$780 for a parking permit
- In 2017, they reportedly accumulated $1 million from permit purchases



PD-19-073Y
Engineering for Civil Infrastructure
National Science Foundation

- “to meet the needs of humans.”
- “radical rethinking of traditional civil infrastructure in response to emerging technological innovations,”

Call To Action

mesaaz
Smart City

WAYMO

Sources

1. Talona Felix- City of Mesa, Time and Labor Administrator
2. ASU Mesa Campus Info Flyer
3. http://zebu.uoregon.edu/disted/ph162/l4.html (solar density)
4. National Safety Council
5. New York Times
6. University of Oregon