ECE 1778:
Creative Applications for Mobile Devices

Lecture 4

January 30, 2019

Today

1. Logistics – Plan, Assignments
   - Description of Assignment S3
   - Proposal
2. Block Diagrams + Living in Apps
3. Notes on Group Work
   - Working in Groups
4. A Note on the Android Life cycle
5. External Sensors/Gadgets you may want to use
6. TeamChooser
   - Lessons from an App development experience
7. Informal Topic Discussions/Coordination

Logistics

Assignments

- S2 was due yesterday
  - Any comments/questions?

- S3 is posted on course website
- P2 is due next Tuesday

- S1 and P1 have been graded
  - Notes from TAs:

Assignment S3

Part 1 Due Monday February 4

Part 2 Due Thursday February 7

All assignments are on both Course website and on Quercus

Assignment S3 – for Specialists

- Recall: one goal of course is to give experience in reaching across disciplines
- Aside: we anticipate that Specialists will be teaching Programmers the **language** of their discipline, and the basic concepts
  - Please, programmers, ask questions – get jargon explained
  - **AND** vice-verse
- Assignment S3 is an attempt to bring Specialists a little into the world of computer engineering/science
  - To give you practice talking to each other

Basic Idea of S3

- Specialist: Choose from one of 5 technical areas listed
  - **that you are not already familiar with,**
  - **and that your programming partners *are* familiar with:**

1. Fast Searching
2. Cloud Server/Frameworks
3. Machine Learning
4. Optimization
5. Internet Communication

Then

First

- Spend an hour with your partners, learning about this area, and take notes.
- **Do Not** use any other sources of information

Part 1

- Write up those notes & submit Monday February 4th, 6pm
- 500 words + pictures
- **Explain in your own words**

Then – Part 2

- Pursue a deeper understanding of the topic, via Internet
- Write another 500 words, due Thursday Feb 7th, 6pm
  - Do a better job of describing the topic; add some nuance & more understanding
- Offer some additional commentary on your view of this learning process
  - how it went
  - how much you learned from Part 1 vs. Part 2,
  - Reflect on what would have made it better
    - In terms of communication and learning process
    - So that you apply this to the project

Project

Project Stages

1. **Forming Groups**
   - Must be formed now
2. **Project Approval-in-Principle**
   - Should be wrapping up
3. **Project Proposal/Plan**
   - Document Due February 11th
4. **Proposal & Plan Presentations**
   - February 13th
   - **NOTE EXTRA Lecture Wed February 13th, 6-8pm, SF 1101**
5. **Spiral 2 & Spiral 4 Presentations**
   - 2: March 6/13 4: March 20/27
6. **Final Presentations**
   - Weeks of April 3/10
7. **Final Report Due April 17th**

Proposal/Plan Due Monday Feb 11 @ 6pm

1. Describe Goal, make more precise
   - What & Why
2. Rough design of what the **user** of the App sees
   - Screen designs
   - Use Marvel App - https://marvelapp.com
     - From Specialist Assignment 2
     - Any drawing package will do
3. **Block Diagram** overview of planned code
   - The large pieces (roughly 5) of the system
     - With short description of each (below picture)
   - Should be linked to the screens
   - I will discuss creation of block diagrams shortly

Plan, continued

4. Statement of Risks/Issues
   - What roadblocks/issues/challenges do you foresee?
   - Software, Hardware, Ethics, Data …
5. What do you need to learn that you don’t know
   - all members
6. **Important:** Specialists
   - Submit a 500 word essay on
     1. How App relates to field of Specialist, and
     2. How the Specialist will contribute to project
        - Need to be an active participant; want clear thought here
          - **Issue Tracking** on Github?

- **Document must have these sections; will lose marks if missing**

Proposal/Plan Document

- length: 1500 words max
  - not including Specialist essay (#6)
  - include word count as part of document, penalty for overage
- Seeking clarity, not quantity of words
  - Omit needles words
- **Specialist** should Submit to Quercus,
  - under Assignment “**Proposal-Plan**”
- Worth 10% of grade
  - including in-class presentation done following week
- Document Due Monday February 11th at 6pm

A Note on Block Diagrams

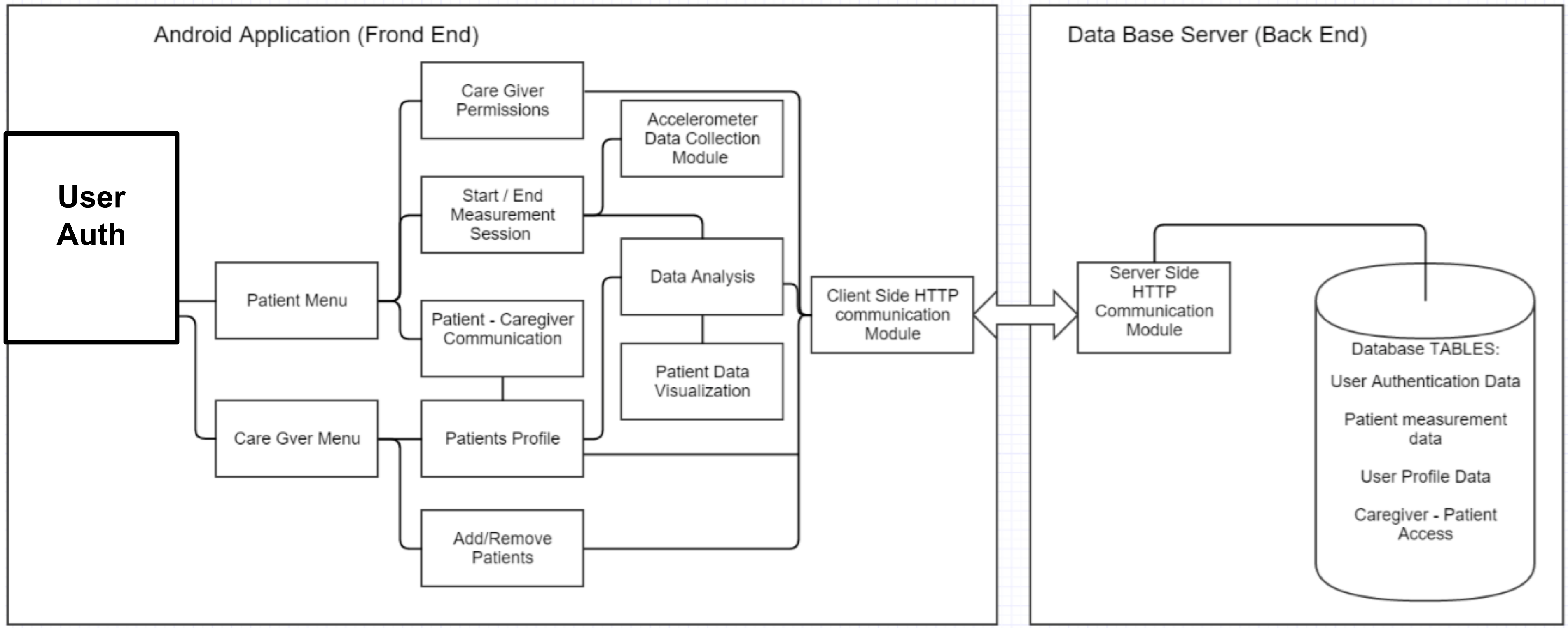
And the creative process

Block Diagrams

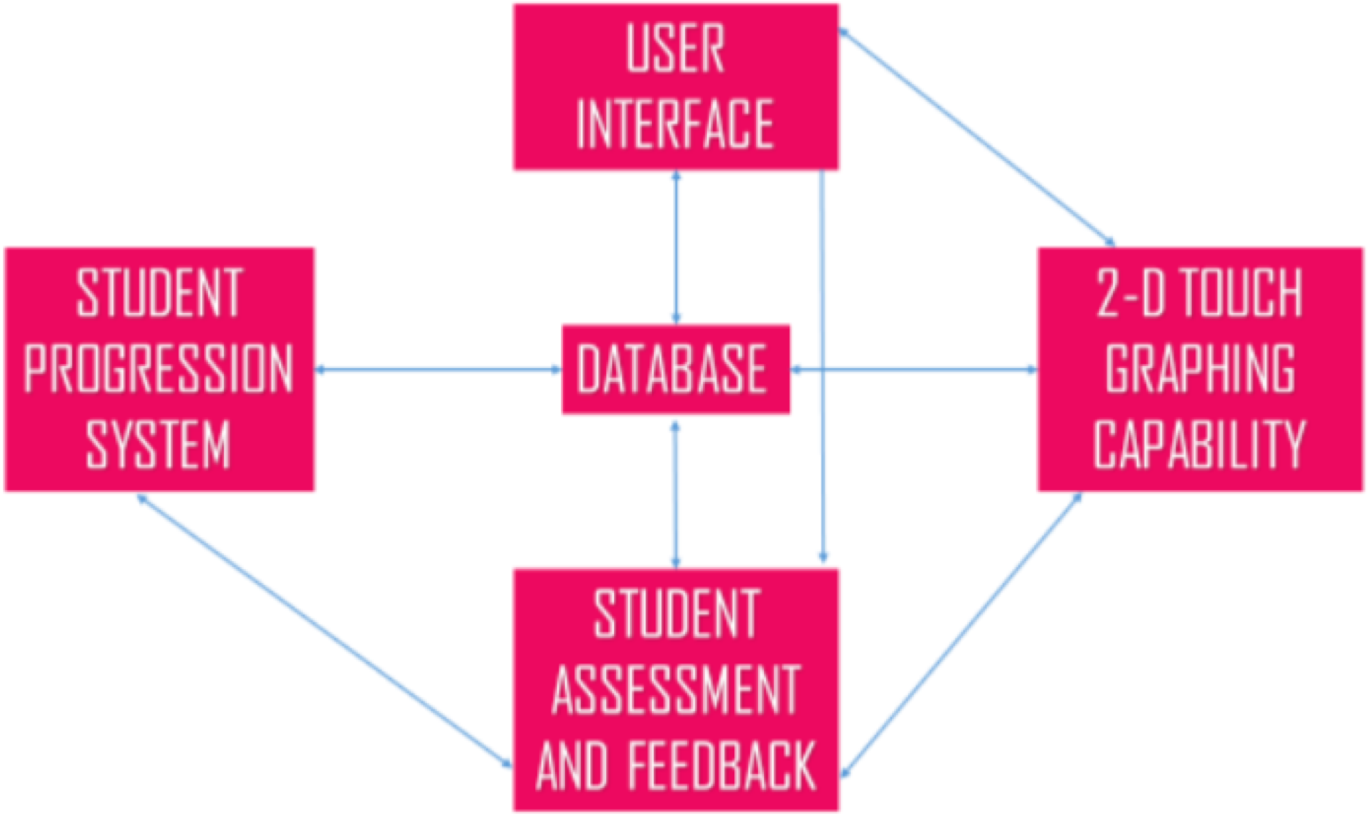
- In Proposal you're asked to give a block diagram of your planned software
  - In past, many have not understood this
- A Block diagram describes the major pieces of the project's functions (roughly 5)
- Is the first step in the standard **divide & conquer** approach
- Draw blocks, give each block a name
  - Name gives a sense of what it does
  - Provide several sentences that give more detail of function of each block

Block Diagrams

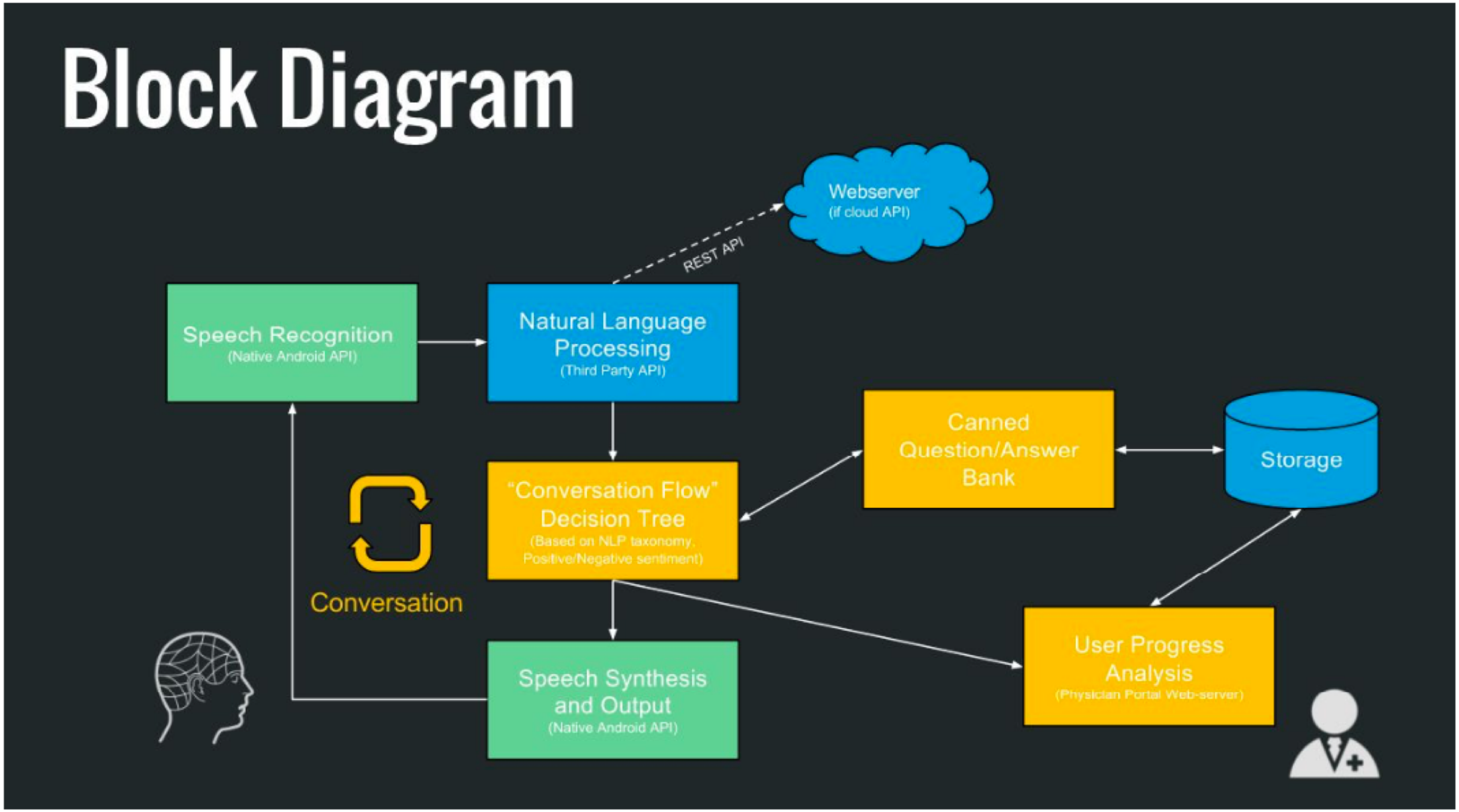
- Lines between blocks show communication
- May need to break blocks themselves down
  - In a hierarchy, in a subsequent part of your plan document
  - This example has **too many** blocks, too small font


Android Application (Frond End)
Data Base Server (Back End)
User Auth
Patient Menu
Care Gver Menu
Care Giver Permissions
Start / End Measurement Session
Patient - Caregiver Communication
Patients Profile
Add/Remove Patients
Accelerometer Data Collection Module
Data Analysis
Patient Data Visualization
Client Side HTTP communication Module
Server Side HTTP Communication Module
Database TABLES:
User Authentication Data
Patient measurement data
User Profile Data
Caregiver - Patient Access

Better, but maybe too simple


USER INTERFACE
STUDENT PROGRESSION SYSTEM
DATABASE
2-D TOUCH GRAPHING CAPABILITY
STUDENT ASSESSMENT AND FEEDBACK

Good



Block Diagram

Webserver
(if cloud API)

REST API

Speech Recognition
(Native Android API)

Natural Language Processing
(Third Party API)

Canned Question/Answer Bank

Storage

"Conversation Flow" Decision Tree
(Based on NLP taxonomy, Positive/Negative sentiment)

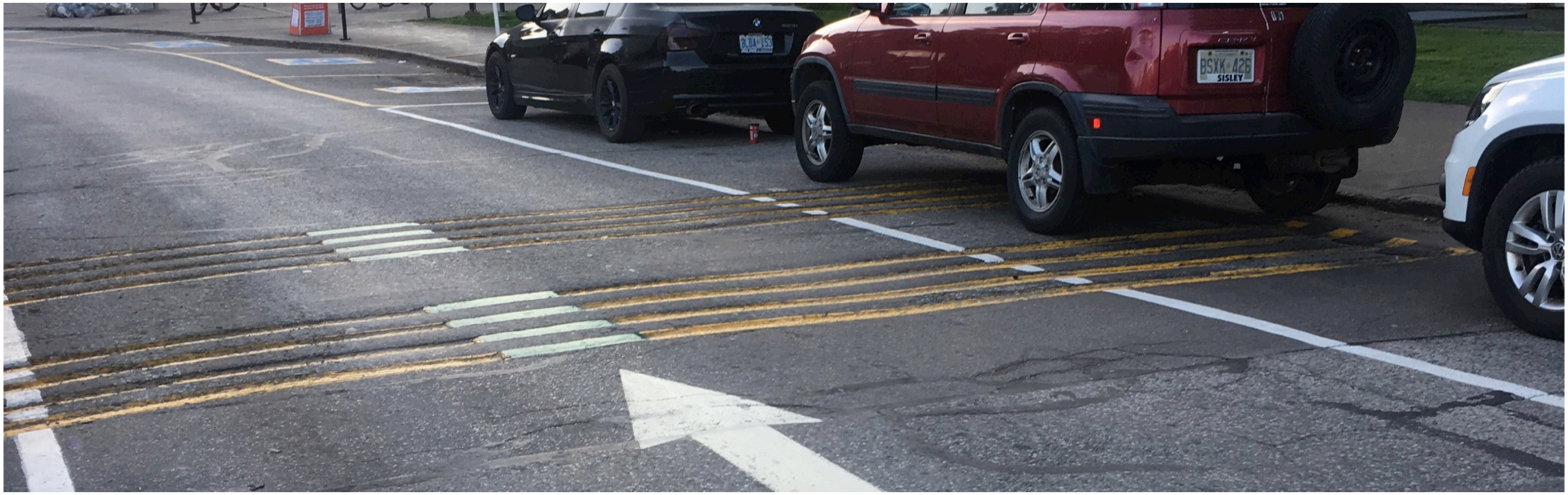
Conversation

Speech Synthesis and Output
(Native Android API)

User Progress Analysis
(Physician Portal Web-server)

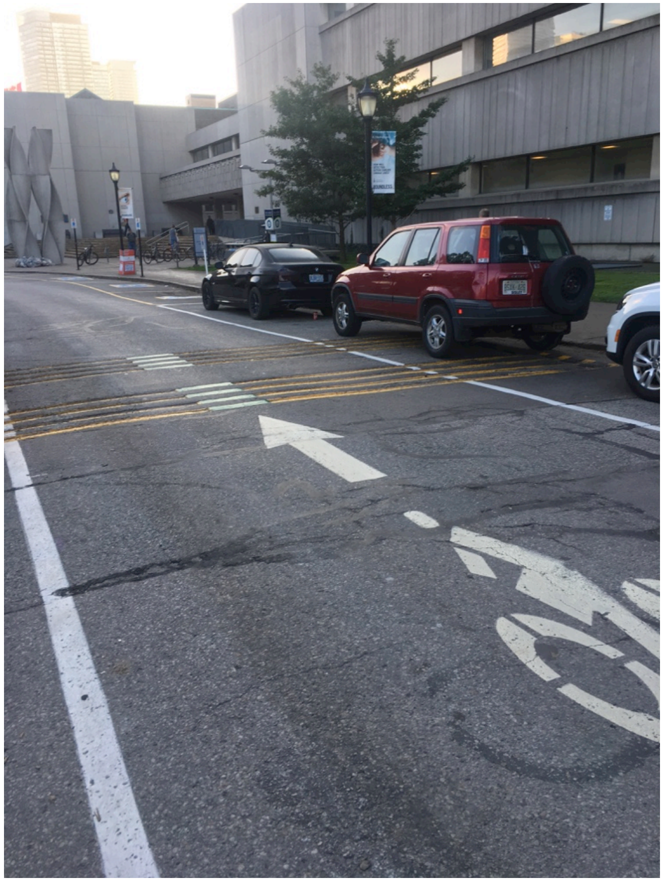
Let’s Work Through An Example

- Problem: Terrible Speed Bumps on King’s College Circle
  - very poorly designed – reverse bumps!
  - hitting the yellow part almost throws bike rider off bike!



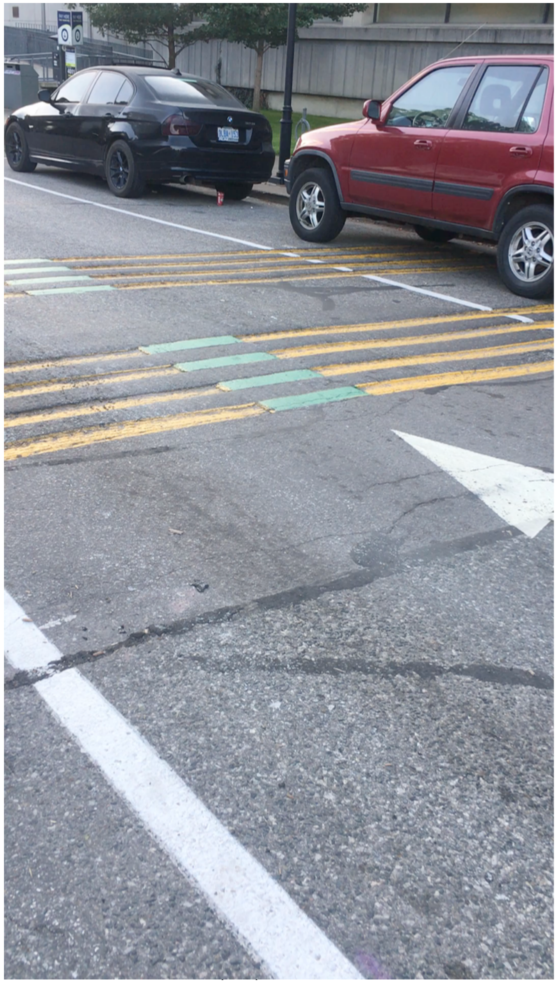
So, Fill in Middle for Smooth Bike Cross

- This forces the cyclist into the middle of the road
- Into the path of any trailing car!



That’s Dumb, But Even Worse

- The speed bumps don’t really seem to affect cars at all!
- Watch:



App Goal:

- Measure Speed Bumps Effectiveness in Car
  - i.e. how noticeable is the bump?
  - Does the bump get worse with higher speed of a car?

- How could you do this with a phone?

Let’s Make the Block Diagram

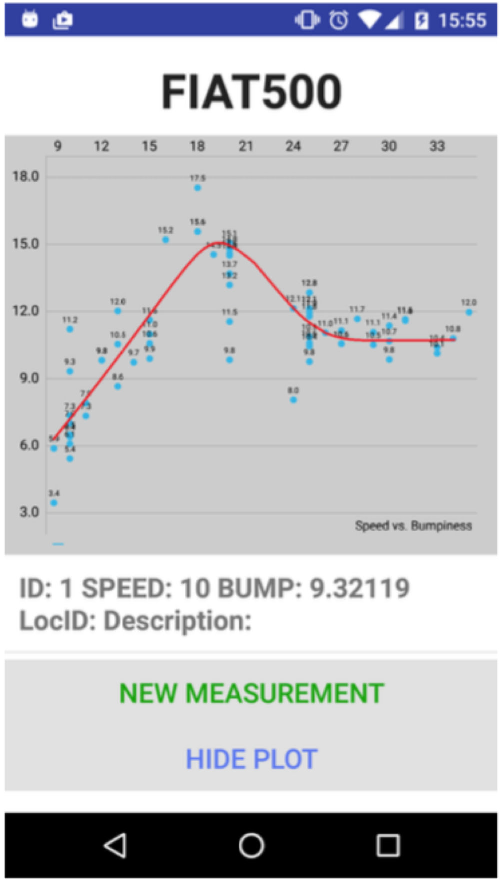
- What are the main components?

Let’s Figure out How to Enhance it

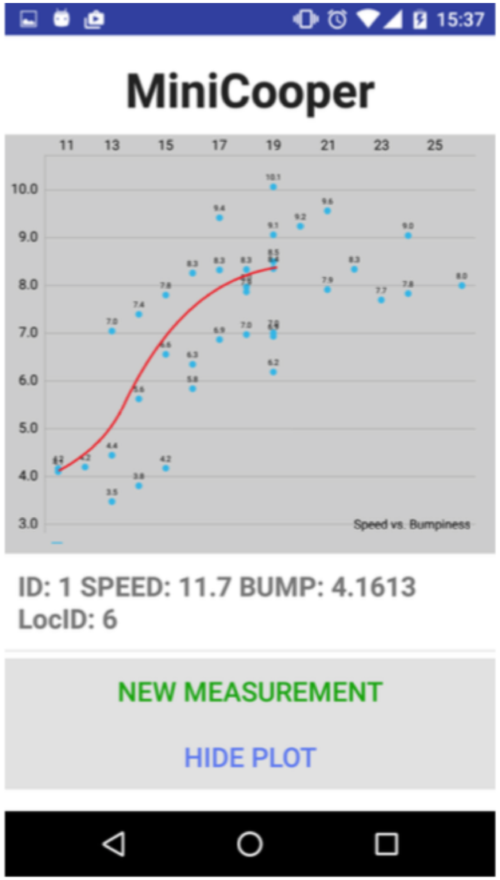
- Think about the context, what could we do to make this more useful/functional/helpful?

Previous Project: “How Bumpy”

- Measurement of Two Different Cars over bumps
  - Did it prove anything?



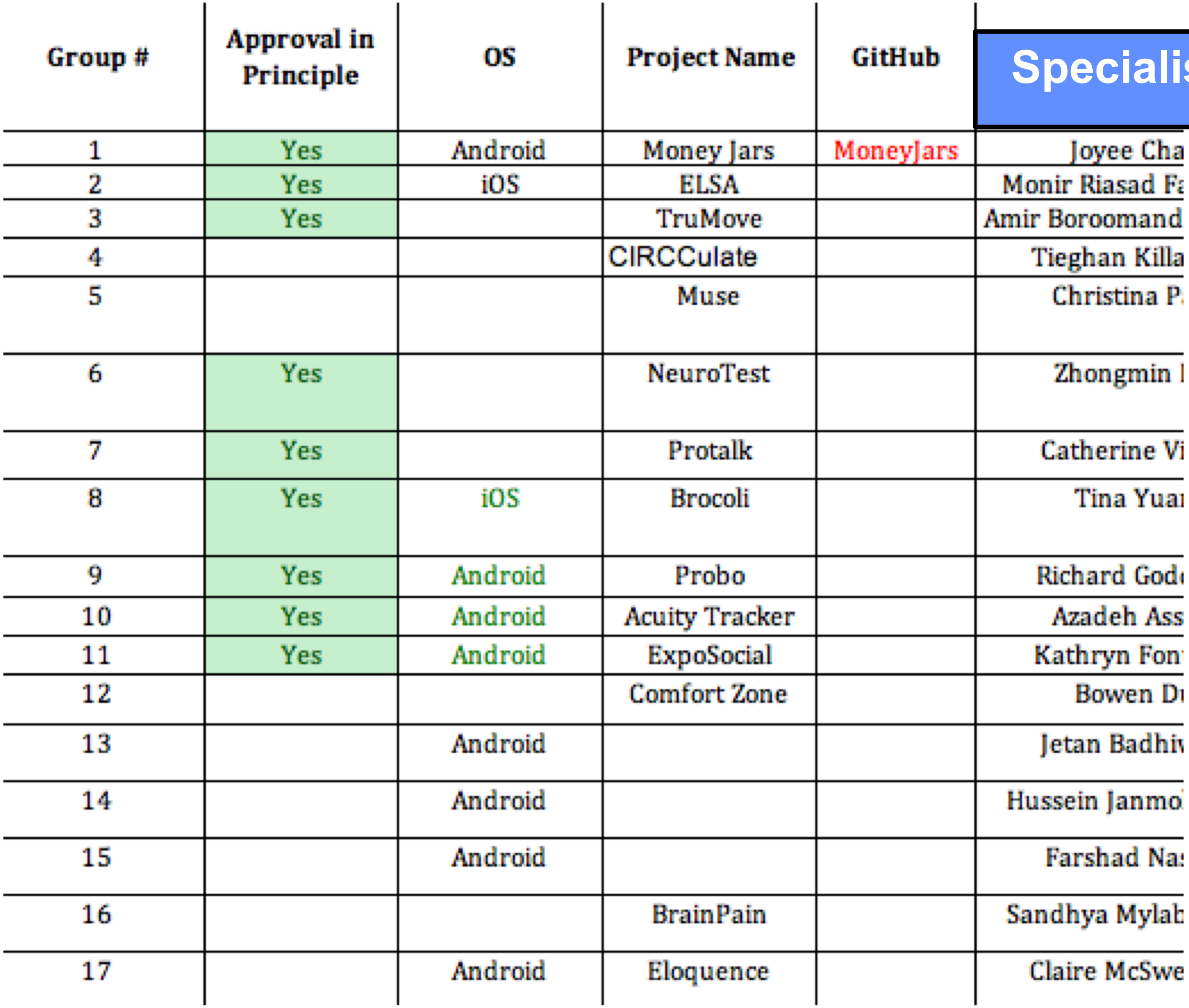
ID: 1 SPEED: 10 BUMP: 9.32119
LocID: Description:



ID: 1 SPEED: 11.7 BUMP: 4.1613
LocID: 6

Project/Groups

Group Names:



Group #	Approval in Principle	OS	Project Name	GitHub	Specialist
1	Yes	Android	Money Jars	MoneyJars	Joyee Cha
2	Yes	iOS	ELSA		Monir Riasad Fa
3	Yes		TruMove		Amir Boroomand
4			CIRCCulate		Tieghan Killa
5			Muse		Christina P
6	Yes		NeuroTest		Zhongmin
7	Yes		Protalk		Catherine Vi
8	Yes	iOS	Brocoli		Tina Yua
9	Yes	Android	Probo		Richard God
10	Yes	Android	Acuity Tracker		Azadeh Ass
11	Yes	Android	ExpoSocial		Kathryn Fon
12			Comfort Zone		Bowen D
13		Android			Jetan Badhi
14		Android			Hussein Janmo
15		Android			Farshad Na
16			BrainPain		Sandhya Mylab
17		Android	Eloquence		Claire McSwe

A Note on Group Interaction

Group Interaction

- Now that groups are formed, it is very important for you to meet regularly and coordinate your activities
- Each of you will need to be assigned tasks, and make commitments to do those tasks at a particular deadline
- The expectations that each group member has of the others must be made clear
  - I suggest that these be written down
  - **Each** meeting in which tasks are decided:
  - Tasks should be recorded at end of each meeting
    - What
    - Who
    - When

“When” is a guess

- **Just** because you thought you could get something done, doesn’t mean you will be able to
- Tasks deadlines may have to ‘slip’ because of under-estimation
- is reality of engineering and indeed all work
  - A key opportunity to learn how to estimate work!
- Group must be forgiving of missed tasks
  - Must communicate and re-organize goals

Difficulties Can and Do Happen

- Most of the difficulties that have occurred in this course have been because expectations were not made clear
- Sometimes, group members consistently failed to live up to commitments.
- If this happens, please report it to me as soon as it is something that you cannot handle with internal discussion & resolution
  - We will help you resolve it
- Disagreements are part of all human relationships
  - One distinguishes oneself by how well you deal with them

What is a good way to communicate?

- Your favourite messaging app
  - Kik, FB Messenger, WeChat….
- Piazza
- Email
- Skype/Google Hangouts

That’s About What Could Go Wrong

- What about doing things to make it go right?
  - See above about basic organization

- What is a good way to create a successful group?
  - Get to know each other – find out goals and interests
  - ‘Break Bread’ have some meals together – breakfast lunch or dinner.
  - Practice the notion of listening and giving everyone’s ideas a fair hearing.

GitHub Requirement

- We now require you to use GitHub for storing software and tracking issues
  - Aside: what is source code control?
  - Demo issue tracker
- I will create the repository for you, and make **all** group members collaborators
  - Including specialists
  - Repository will be private
- Your immediate task, by this Friday:
  - Send me your GitHub ID so I can connect you to the repo
  - If you don’t have one, go to **github.com** and sign up
    - Then send your GitHub to me with group name
    - I will link you to the repository

Class Participation & Peer Review

Class Presentations

- A key part of what happens in this course is the contribution you make to other's projects

- You will do many presentations in this class
  - Indeed, one side-effect of this project course is some real practice in giving high-quality, concise & clear communication
  - Most presentations will be 5 minutes in length
  - Must be geared so that most people in the class will understand

Peer Review

- Want everyone to come, listen & provide useful feedback
- Expectation that you’ll listen and provide thoughtful feedback and suggestions to other’s presentations

In Addition

- For each of Proposal, Spiral 1, and Spiral 2 you’ll be asked to provide a written peer review for one other group; these will be graded
- Means you’ll need to be here for every lecture, not just when you’re presenting.

Peer Review for Proposal

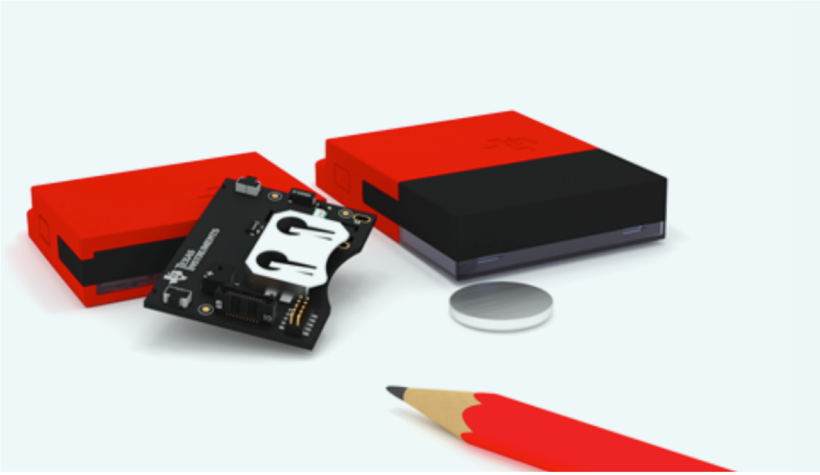
Will be asked to answer questions such as:

1. In your own words, what was the goal of the project?
2. What parts of the proposal did you understand, and what parts could be more clear?
3. What was the best thing about the proposal?
4. What one thing could be improved the most?

External Gadgets

Texas Instruments Sensor Tag 2

- **Bluetooth Connection**
- **Sensors:**
  - 9 axis
  - Magnet sensor
  - Light
  - Ambient temperature
  - IR temperature
  - Humidity
  - Air pressure
  - Two Buttons, two lights, quiet buzzer!
- I can order these as needed
- Demo



Muse Head Band

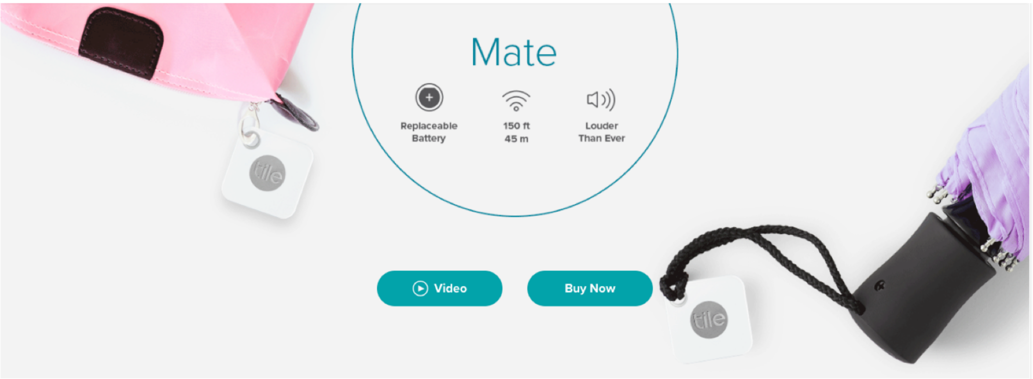
- Measures ‘brain activity’ through measurement of electro-magnetic waves brain produces
- Have 1 of these

TILE Tracker

- Note where things are, where they were lost!
  - https://www.thetileapp.com
  - https://youtu.be/WG7BdW7iFzo
  - Don’t have any, but could order
  - There is an unofficial API documented/coded here:
  - https://github.com/bachya/pytile

tile TILE now

It looks like you left home without your ‘Keys’.



Other External Devices I have

- Wahoo Tickr (1) – heart rate monitor strap
  - bluetooth connected
  - Direct live heart rate

Other Sensors/Gadgets?

- If you find one that will enable your project, you can request to have it purchased
  - I try to collect these for this course and research
  - No guarantee, depends on cost & function, but ask if you see something interesting!

Small Programming Note:
The Android Life Cycle

Android Application Life Cycle

- Recall: Activities are screens that the user sees, and associated process
- Android manages these Activities as a **stack**.
- When a new activity is started, it is placed on the top of the stack and becomes the running activity

- The previous activity always remains below it in the stack,
  - and will not come to the foreground again until the new activity exits.

Important to Pay Attention to 'LifeCycle'

- To ensure app behaves well in several ways, including:

1. Does not crash if the user receives a phone call or switches to another app
2. Does not consume valuable system resources when the user is not actively using your app
3. Does not lose the user's progress if they leave your app and return to it at a later time
4. Does not crash or lose the user's progress when the screen rotates between landscape and portrait orientation.

An Activity Can Be in 1 of 4 ‘States’

State 1: Active/Running

- Activity in the foreground of the screen (at the top of the stack)
- Has ‘focus’, meaning user interactions go to it.

State 2: Paused

- activity has lost focus but is still visible
- a new smaller or transparent activity has focus on top of the activity)
- A paused activity is completely alive (it maintains all state and member information and remains attached to the window manager), but can be killed by the system in extreme low memory situations.

Activity States 3 and 4

State 3: Stopped

- activity is completely obscured by another activity
- retains all state and member information
- no longer visible to the user so its window is hidden
- it will often be killed by the system when memory is needed elsewhere.

State 4: Destroyed

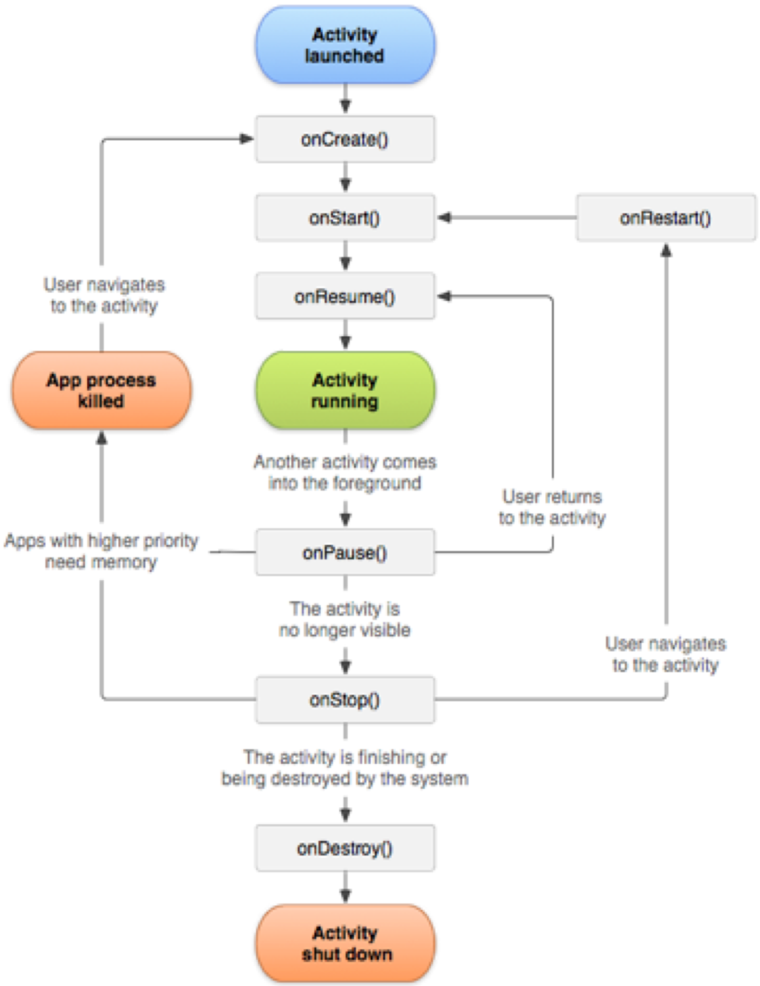
- If an activity is paused or stopped, the system can drop the activity from memory by either asking it to finish, **or simply killing its process.**
- When displayed again to the user, it must be completely restarted and restored to its previous state.

Android Talking to Your App

- The Android operating system asks (or tells) your app to go into those different states by invoking methods associated with your Activity

Methods Called By Android to Change States

- Diagram shows states and methods called to change state
  - Colours: the states
  - https://developer.android.com/guide/components/activities/activity-lifecycle



Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

Three Key States

- Activity can be in 1 of 3 states for long period of time:

1. Resumed
   - In this state, the activity is in the foreground and the user can interact with it. (Also sometimes referred to as the "running" state.)
2. Paused
   - In this state, the activity is partially obscured by another activity—the other activity that's in the foreground is semi-transparent or doesn't cover the entire screen. The paused activity does not receive user input and cannot execute any code.
3. Stopped
   - In this state, the activity is completely hidden and not visible to the user; it is considered to be in the background. While stopped, the activity instance and all its state information such as member variables is retained, but it cannot execute any code.

State Management

- The other states (Created and Started) are transient and the system quickly moves from them to the next state by calling the next lifecycle callback method. That is, after the system calls onCreate(), it quickly calls onStart(), which is quickly followed by onResume().

- Depending on the complexity of your activity, you probably don't need to implement all the lifecycle methods.
- However, it's important that you understand each one and implement those that ensure your app behaves the way users expect.

References

1. The Android Documentation:

https://developer.android.com/guide/components/activities/activity-lifecycle

2. Murphy, Busy Coder’s Android, Chapter “Activities and their Lifecycles” (Page 316)

- Once your project gets going, it is really important to read through this and understand it
  - Previous years’ students pointed out that this was the key thing they had not understood in Android, that caused the most problems

The Key ‘LifeCycle’ Methods

OnCreate()

- Familiar with already – brings the activity to life

OnPause()

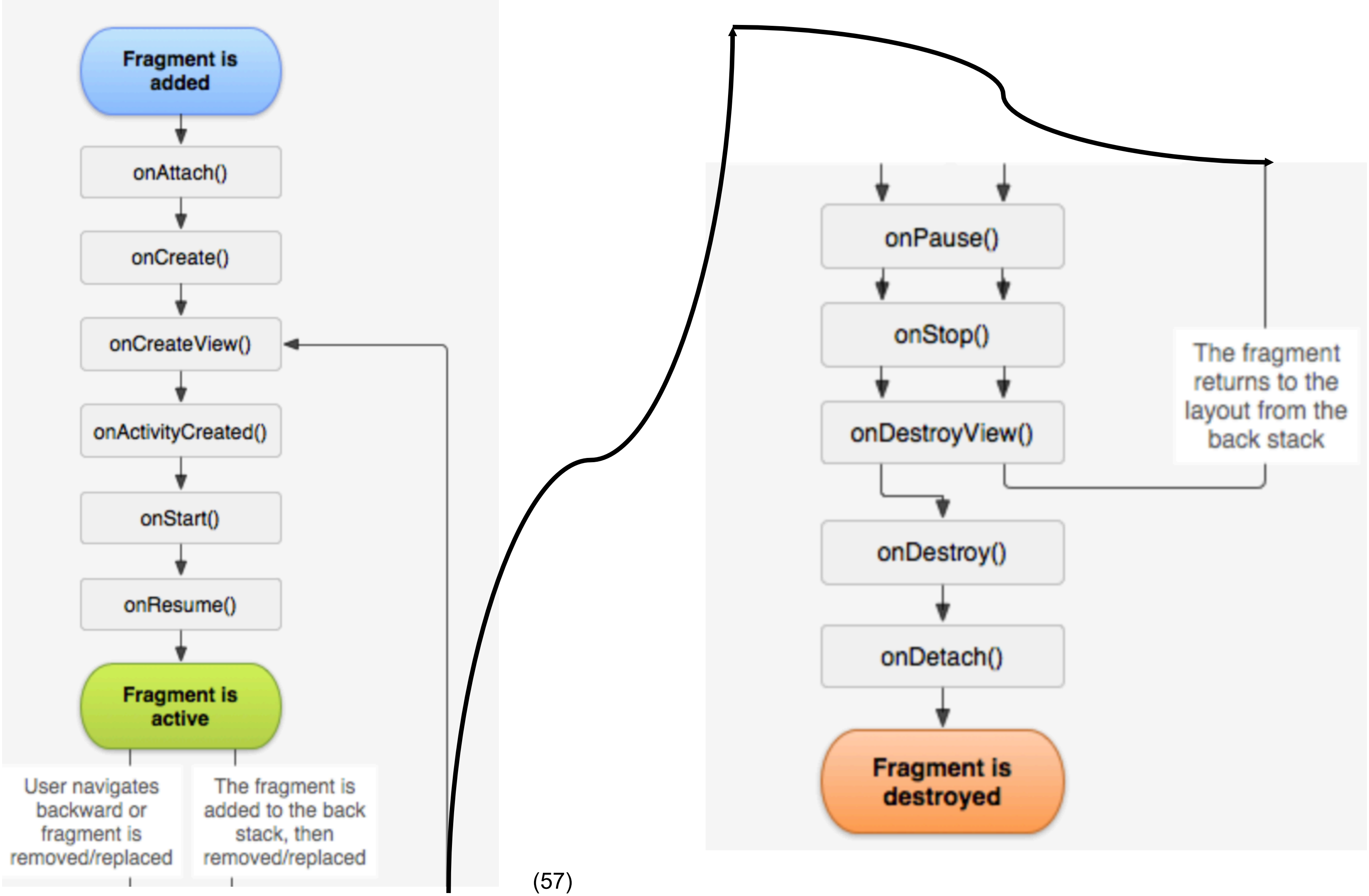
- Another Activity has gained the ‘focus’
- Should stop any background threads, release large resources (such as a camera)
- **No guarantee that OnDestroy() will be called**, so best to save **all** state here

OnResume()

- Called as activity starts, **or** is restarted from a pause
- Can recall state from file, refresh the User Interface – see example

Fragments Behave Similarly

http://developer.android.com/guide/components/fragments.html



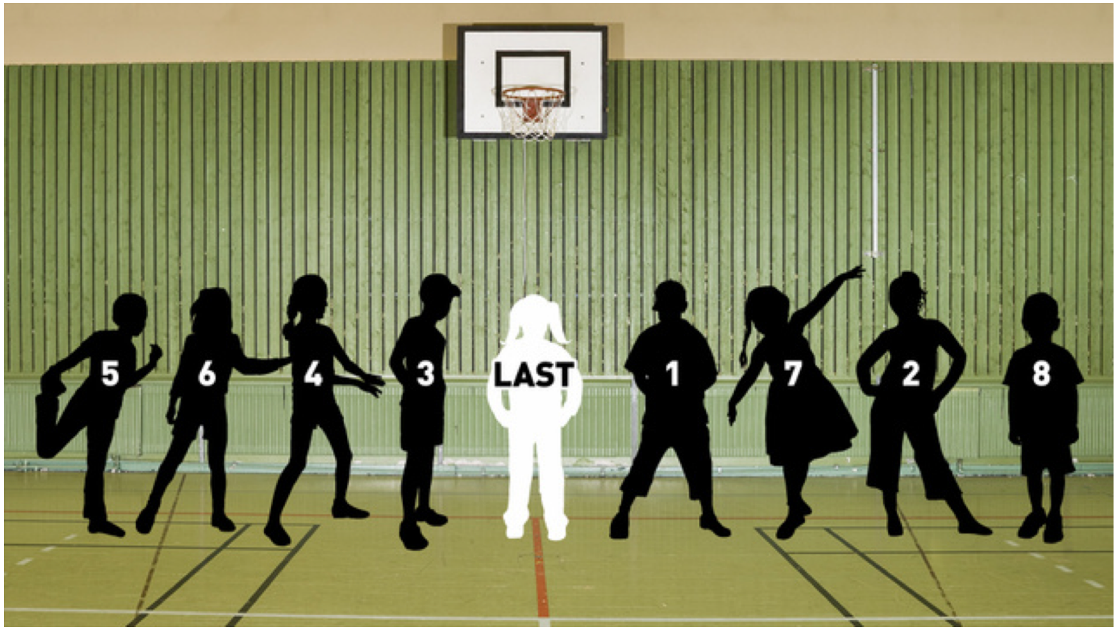
My App: TeamChooser

Solving a Problem in Pick-up Team Sports

Kids Who Play Want Games to be Fair!

So Do Adults!

- When playing friendly games there is a need to choose who is on which team
- A common method is to have team captains, and they alternate choosing people, in a very public way
- How many people have been picked **first?** ☺
- How many people have been picked **last?** ☹



I’ve Been Playing Hockey for Many Years

- A friendly game, but still have problem choosing teams:
- Classic Canadian method:
  - Put players’ sticks into middle
  - One person randomly throws sticks to either side!
  - Random outcome!

- I once chose teams for a few years in friendly game
  - People complained a lot!

Play in Two Different Friendly Games

- **Wednesday** Game: terrible chooser (**Agar**)
  - People always complaining
  - Games often lopsided, much distaste
  - no-one else took over, though (didn't want the hassle?)

- **Sunday** Game: excellent chooser (**Paul**)
  - Paul had a natural ability to pick great teams!
  - Even when teams didn't look right, many more times than not, the game was fair
  - Became known as the '**algorithm**'

The Solution: TeamChooser

- Wouldn't it be great if an App made the teams?
  - No one to yell at
  - Possibly give better teams

- Who needs this?

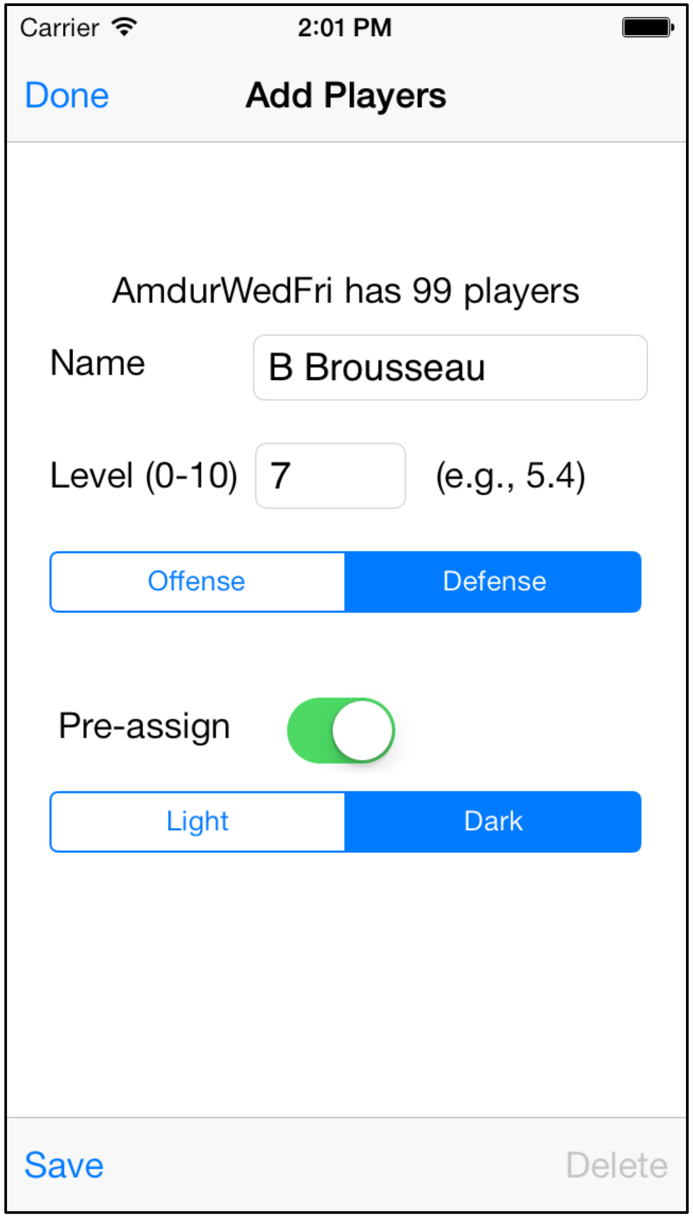
- Every pick-up hockey, soccer, basketball game around!

TeamChooser: How It Works

- Enter every player in advance of game day
  - Player's name
  - Preferred position (offence or defense)
  - **A rating, from 1-10, as to how effective player is**
    - Rating is the trickiest part

- On game day – select all players present
- Push '**Make Teams**'
  - And voila, two evenly matched teams

Do Demo! Entering Players



AmdurWedFri has 99 players

Name: B Brousseau

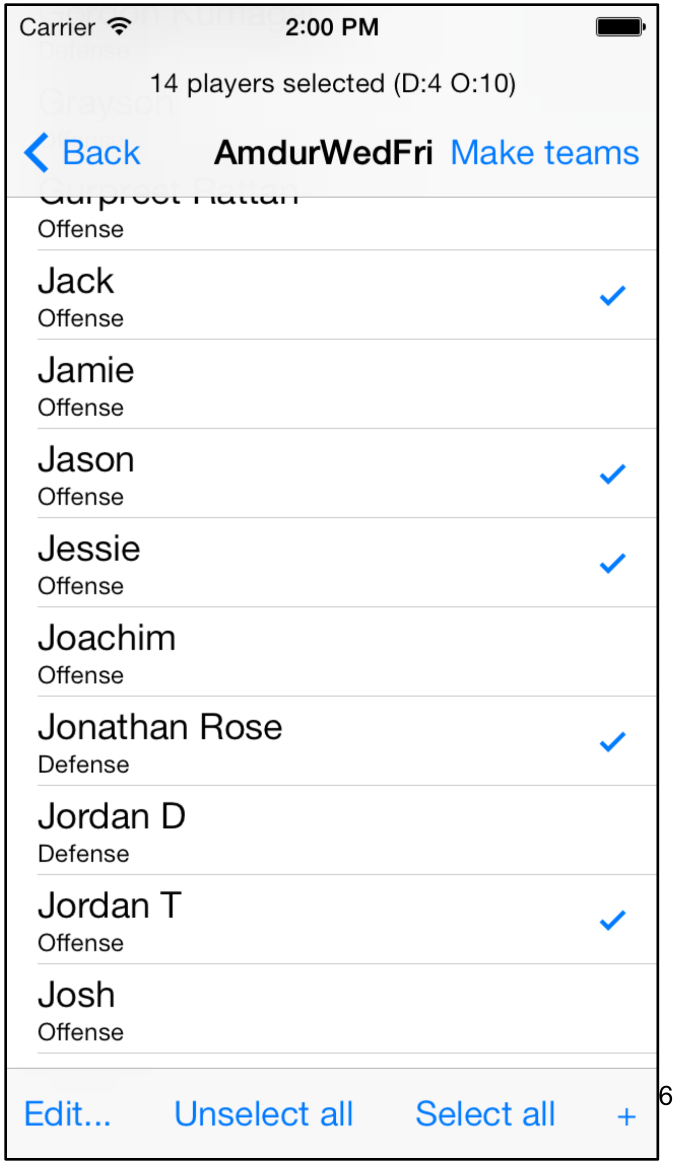
Level (0-10): 7 (e.g., 5.4)

Offense [✓ Defense]

Pre-assign: ● on

Light [✓ Dark]

Selecting Present & Making Teams



Carrier 2:00 PM

14 players selected (D:4 O:10)

< Back **AmdurWedFri** Make teams

Gurpreet Rattan
Offense

Jack ✓
Offense

Jamie
Offense

Jason ✓
Offense

Jessie ✓
Offense

Joachim
Offense

Jonathan Rose ✓
Defense

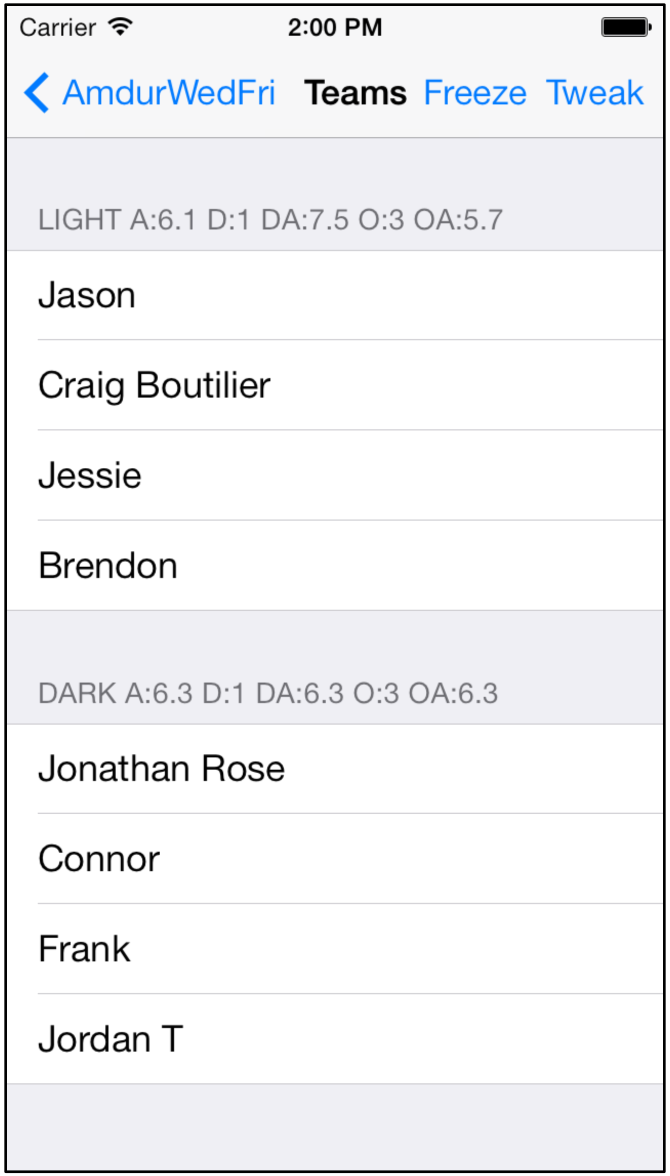
Jordan D
Defense

Jordan T ✓
Offense

Josh
Offense

Edit... Unselect all Select all +

6)



Carrier 2:00 PM

< AmdurWedFri **Teams** Freeze Tweak

LIGHT A:6.1 D:1 DA:7.5 O:3 OA:5.7

Jason

Craig Boutilier

Jessie

Brendon

DARK A:6.3 D:1 DA:6.3 O:3 OA:6.3

Jonathan Rose

Connor

Frank

Jordan T

Example Game

Dark Team

DARK 5.6 D:5.0 O:5.7 COUNT: 6

Cole Zem

Brian Dov

Pavel

Raj B

Matt Zale

Rich Zemei

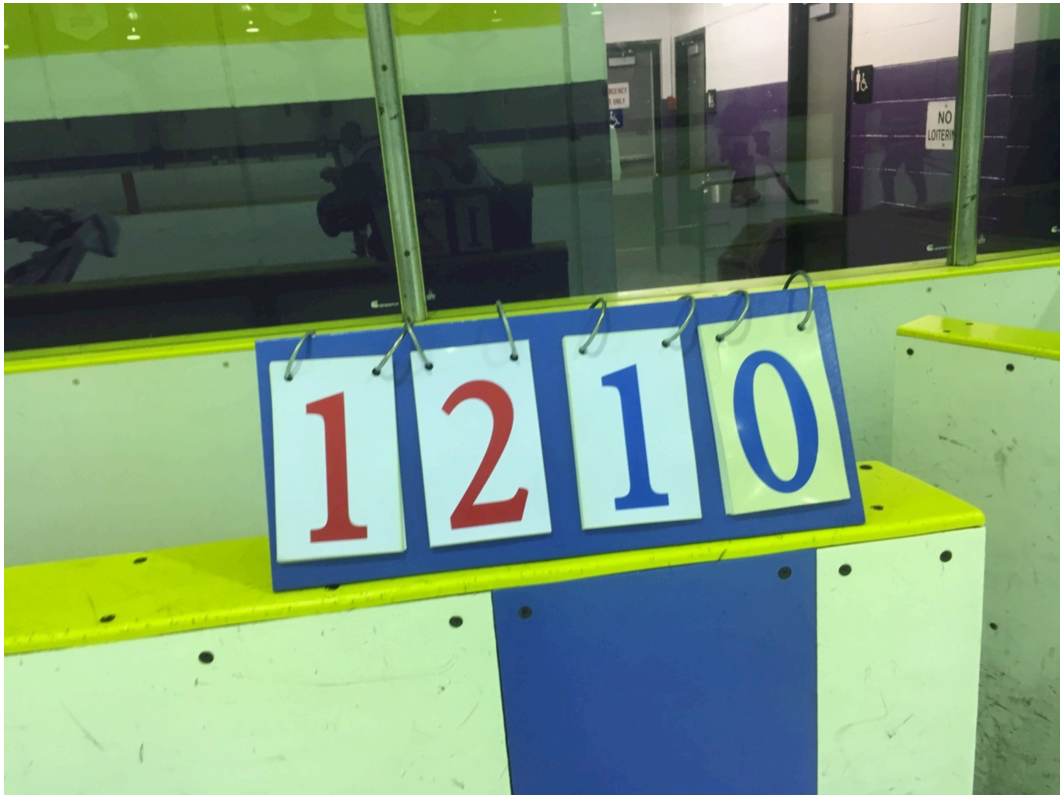
Light Team

LIGHT 5.6 D:5.4 O:5.7 COUNT: 7

Gurpreet Rattan

Final Score in that Game

- A Victory for Team Chooser!



Team Selection Method (‘Algorithm’)

Step 1: Sort Players in Order of Rating, Highest to Lowest

Step 2: Alternate Team Assigned Going Down List

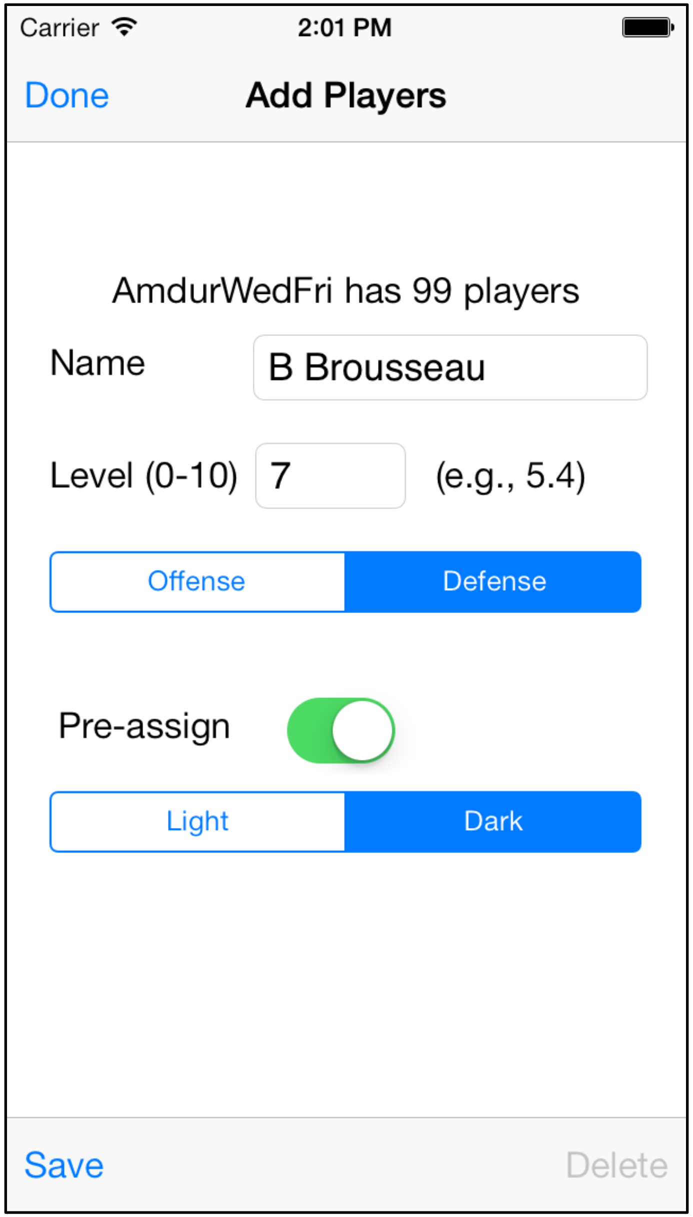
Team A		Team B
	6	
	5	
	4	
	3	
	2	
	1	

Team Selection Method

- Gets more complicated when add features!
  - Balance offense and defensive player count
  - Pre-assigns
  - Balance quality of offense and defense

- Many discussions from CS and ECE Professors over algorithms in hockey game!

Entering Players



Add Players

AmdurWedFri has 99 players

Name B Brousseau

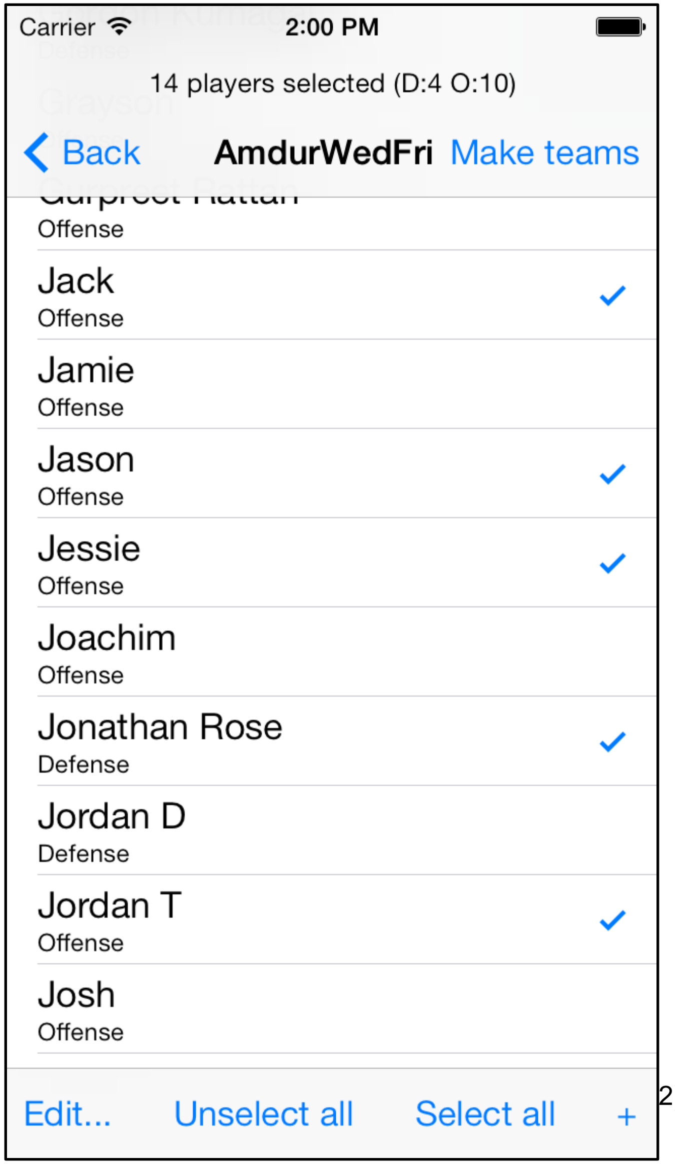
Level (0-10) 7 (e.g., 5.4)

Offense [✓ Defense]

Pre-assign

Light [✓ Dark]

Selecting Present & Making Teams



Carrier 2:00 PM

14 players selected (D:4 O:10)

< Back **AmdurWedFri** Make teams

Gurpreet Rattan
Offense

Jack ✓
Offense

Jamie
Offense

Jason ✓
Offense

Jessie ✓
Offense

Joachim
Offense

Jonathan Rose ✓
Defense

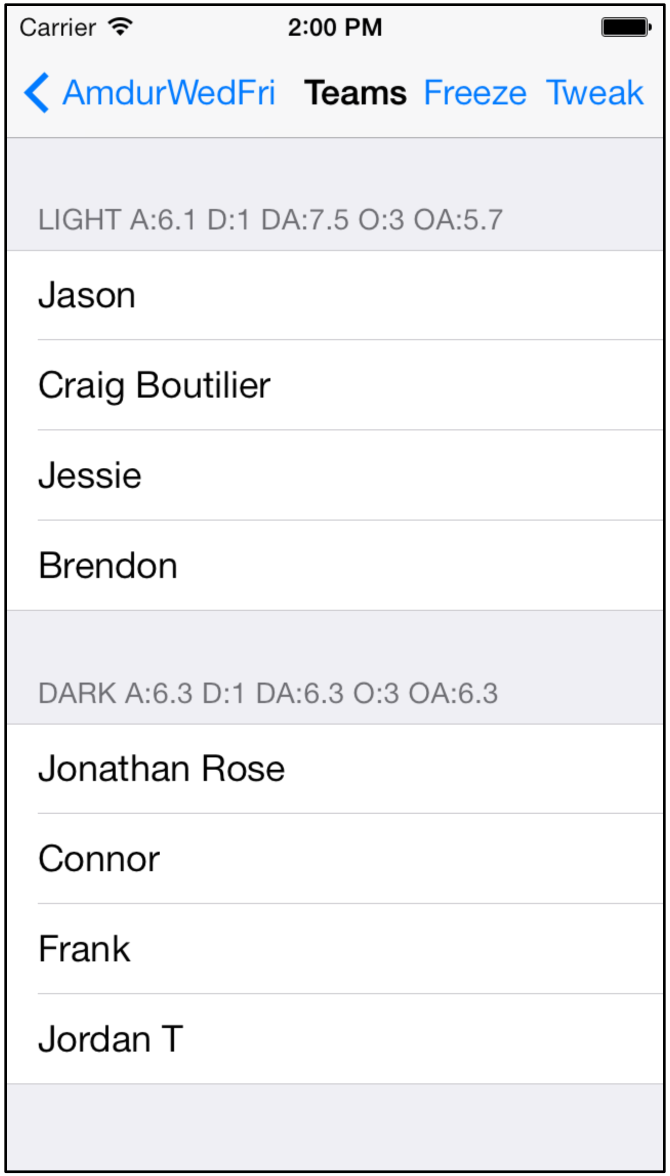
Jordan D
Defense

Jordan T ✓
Offense

Josh
Offense

Edit... Unselect all Select all +

2)



Carrier 2:00 PM

< AmdurWedFri **Teams** Freeze Tweak

LIGHT A:6.1 D:1 DA:7.5 O:3 OA:5.7

Jason

Craig Boutilier

Jessie

Brendon

DARK A:6.3 D:1 DA:6.3 O:3 OA:6.3

Jonathan Rose

Connor

Frank

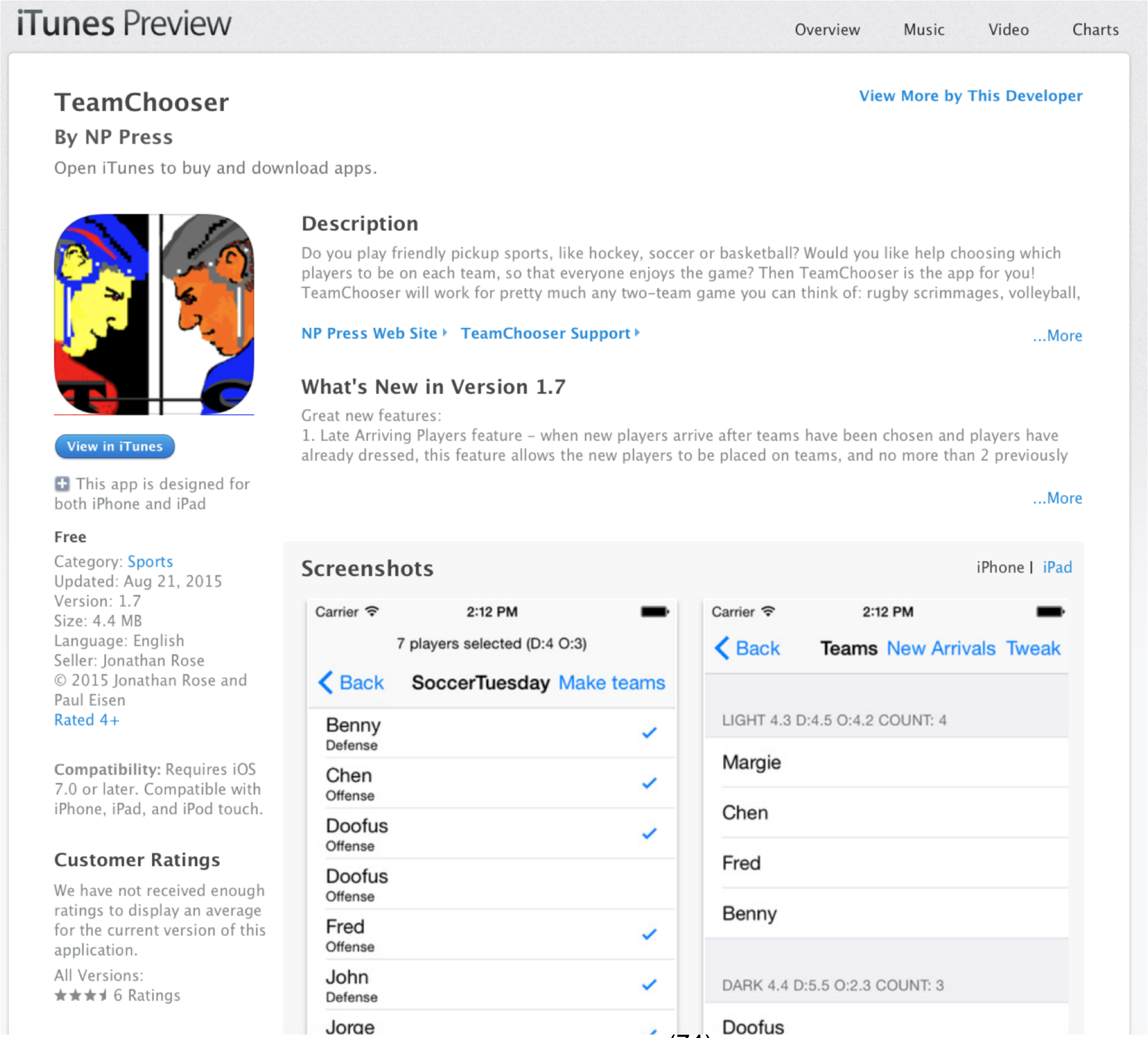
Jordan T

Does it Work?

- Yes!

- I’ve been using it with friends in roughly 600 hockey games and it has often done a good job.
  - We’ve tweaked it’s algorithms here and there
  - Added some features
  - Occasionally very unbalanced games, bad luck?

On iPhone App Store Since May 2010



iTunes Preview

Overview Music Video Charts

TeamChooser

By NP Press

Open iTunes to buy and download apps.

View More by This Developer

View in iTunes

This app is designed for both iPhone and iPad

Free

Category: Sports
Updated: Aug 21, 2015
Version: 1.7
Size: 4.4 MB
Language: English
Seller: Jonathan Rose
© 2015 Jonathan Rose and Paul Eisen
Rated 4+

Compatibility: Requires iOS 7.0 or later. Compatible with iPhone, iPad, and iPod touch.

Customer Ratings

We have not received enough ratings to display an average for the current version of this application.

All Versions:
★★★½ 6 Ratings

Description

Do you play friendly pickup sports, like hockey, soccer or basketball? Would you like help choosing which players to be on each team, so that everyone enjoys the game? Then TeamChooser is the app for you! TeamChooser will work for pretty much any two-team game you can think of: rugby scrimmages, volleyball,

NP Press Web Site ▸ TeamChooser Support ▸ ...More

What's New in Version 1.7

Great new features:
1. Late Arriving Players feature – when new players arrive after teams have been chosen and players have already dressed, this feature allows the new players to be placed on teams, and no more than 2 previously

...More

Screenshots

iPhone | iPad

Carrier 2:12 PM

7 players selected (D:4 O:3)

< Back SoccerTuesday Make teams

Benny ✓
Defense
Chen ✓
Offense
Doofus ✓
Offense
Doofus
Offense
Fred ✓
Offense
John ✓
Defense
Jorge

Carrier 2:12 PM

< Back Teams New Arrivals Tweak

LIGHT 4.3 D:4.5 O:4.2 COUNT: 4

Margie
Chen
Fred
Benny

DARK 4.4 D:5.5 O:2.3 COUNT: 3

Doofus

- Free!
- 4000 Downloads
- Mostly in US/Canada, but a few in UK, Ireland, Japan, Norway, Romania, Portugal, Australia, Denmark, Finland

Lesson Learned

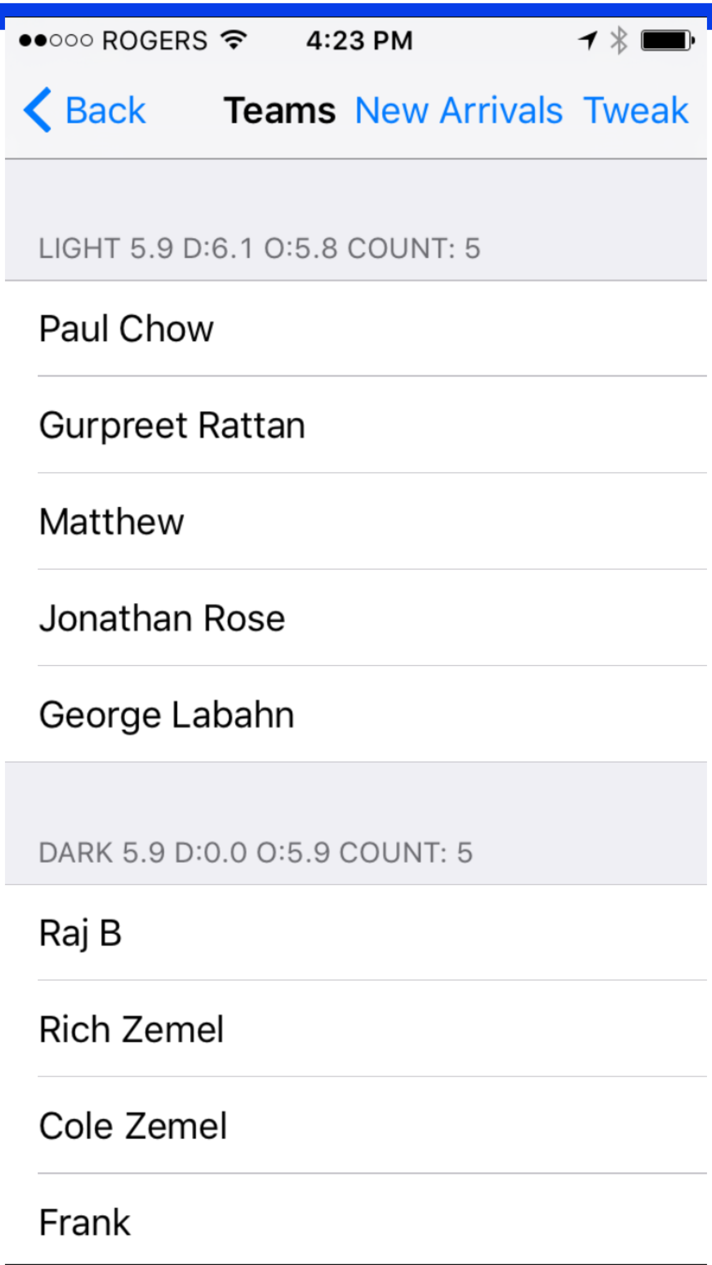
- From last time: Ratings of players, key part of engine, **must be kept secret!**
- Can add password to App so no-one can grab phone and look at their ratings or anyone else’s

Sad Outcome: Agar

- The day after the first use of TeamChooser on Wednesday
  - Recall very poor chooser – Agar
  - Folks joked and called it the iGar.
- Agar **never played again on Wednesday!**
  - Still plays hockey, but not with team he played with for decades

Feature: No-One Chosen Last!

- The order that people are listed (and spoken out loud by user) is randomized
- There is no way to infer rating



LIGHT 5.9 D:6.1 O:5.8 COUNT: 5

Paul Chow

Gurpreet Rattan

Matthew

Jonathan Rose

George Labahn

DARK 5.9 D:0.0 O:5.9 COUNT: 5

Raj B

Rich Zemel

Cole Zemel

Frank

Improvements Needed

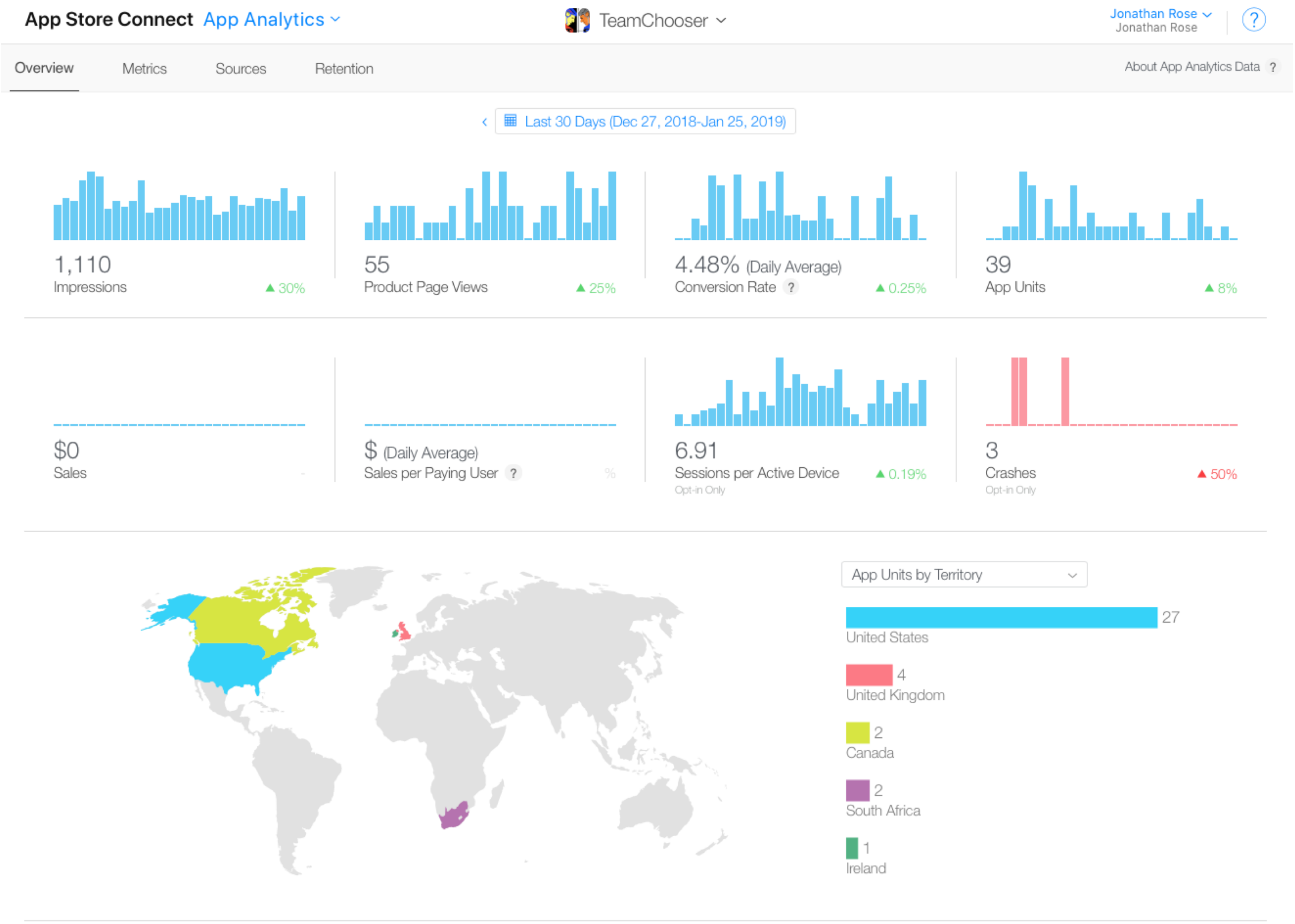
- **Really** needs a backing website
  - To support a business model of advertising, promotions related to sports

- Much discussion about using results of games to determine better ratings
  - Rating players is the most difficult part of using
  - Explored last year as part of personal ML learning

- Don’t really have time to support
  - Have continued to improve the algorithm
  - Added Late Arrival Feature
  - Recently added key stats

Downloads Over the Last 30 Days

- Is delightful to see people using!



Last 30 Days (Dec 27, 2018-Jan 25, 2019)

1,110	Impressions	▲ 30%
55	Product Page Views	▲ 25%
4.48% (Daily Average)	Conversion Rate ?	▲ 0.25%
39	App Units	▲ 8%
$0	Sales	-
$ (Daily Average)	Sales per Paying User ?	%
6.91	Sessions per Active Device Opt-in Only	▲ 0.19%
3	Crashes Opt-in Only	▲ 50%

App Units by Territory

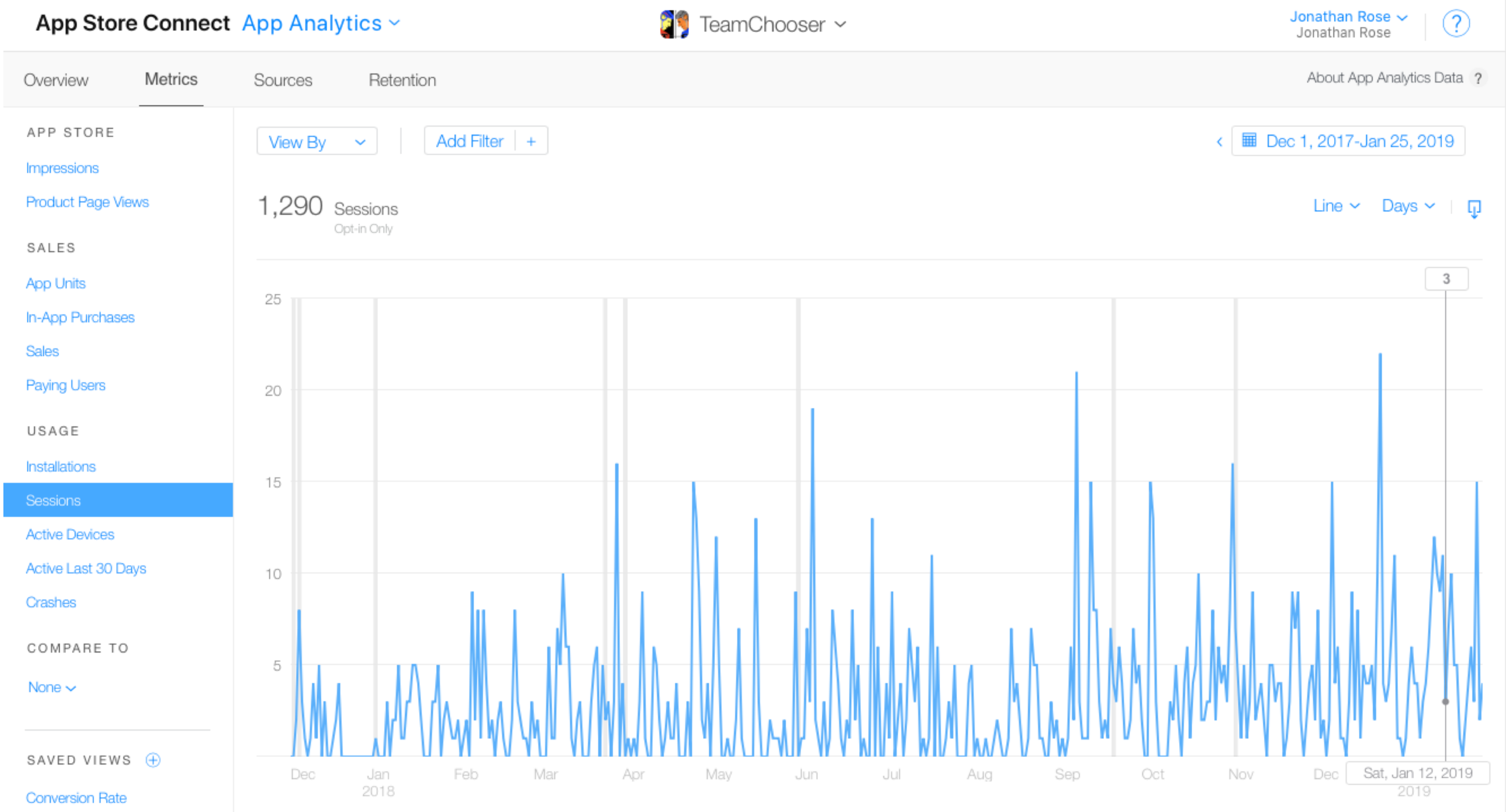
United States	27
United Kingdom	4
Canada	2
South Africa	2
Ireland	1

Is Anyone Using it Who Bought It?

- Instrumented both with Apple Analytics & Firebase lytics
  - Very easy to insert into any app

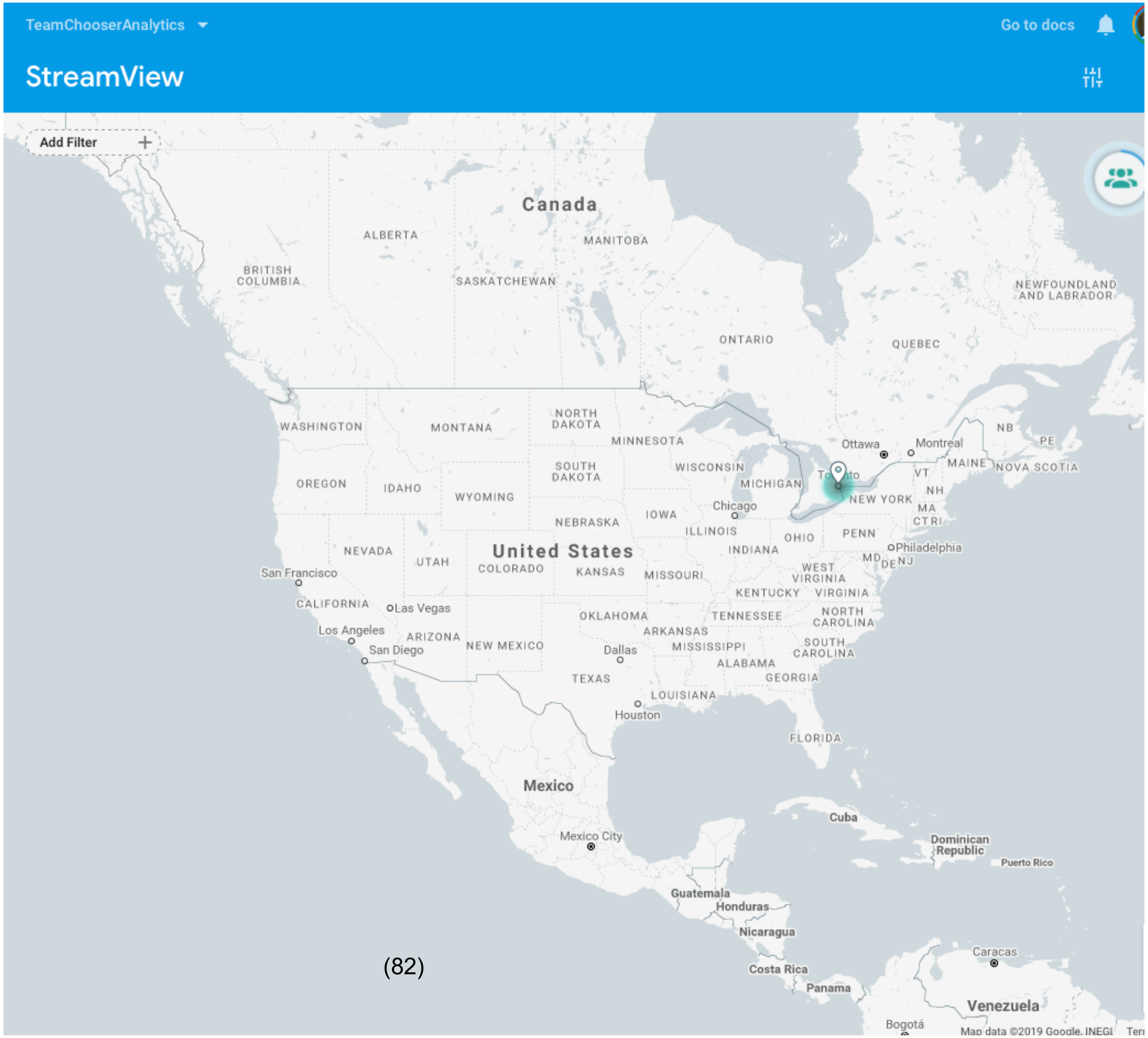
- Reports:
  - # of users sessions, amount of time spent on app
  - Specific pages/events, as you wish from each user
  - Location of user, if already use GPS (no other ID).
  - **Anything I wish to report!**

Session Report from Apple Analytics

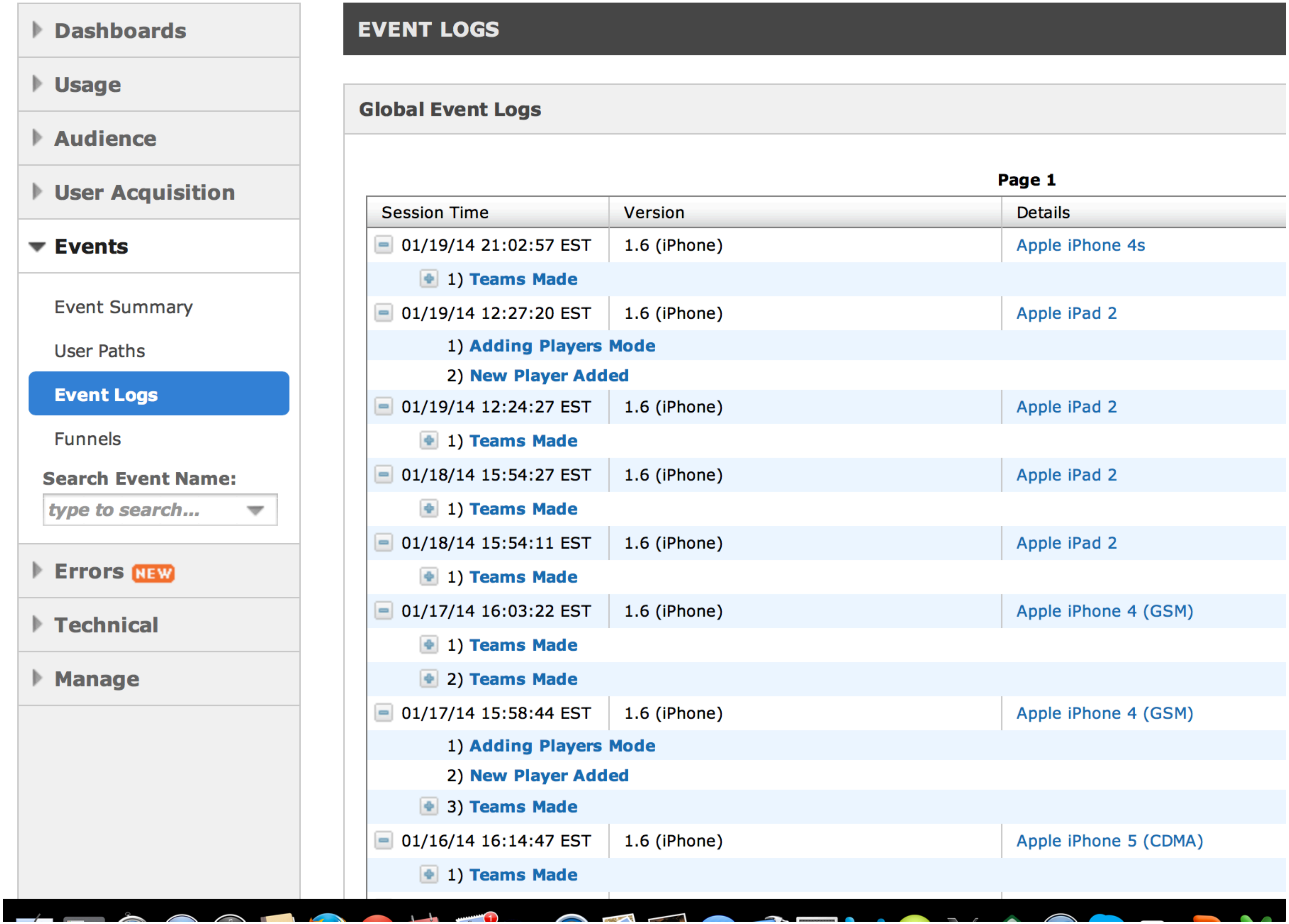


1,290 Sessions
Opt-in Only

Stream View – Live!



Event Logs



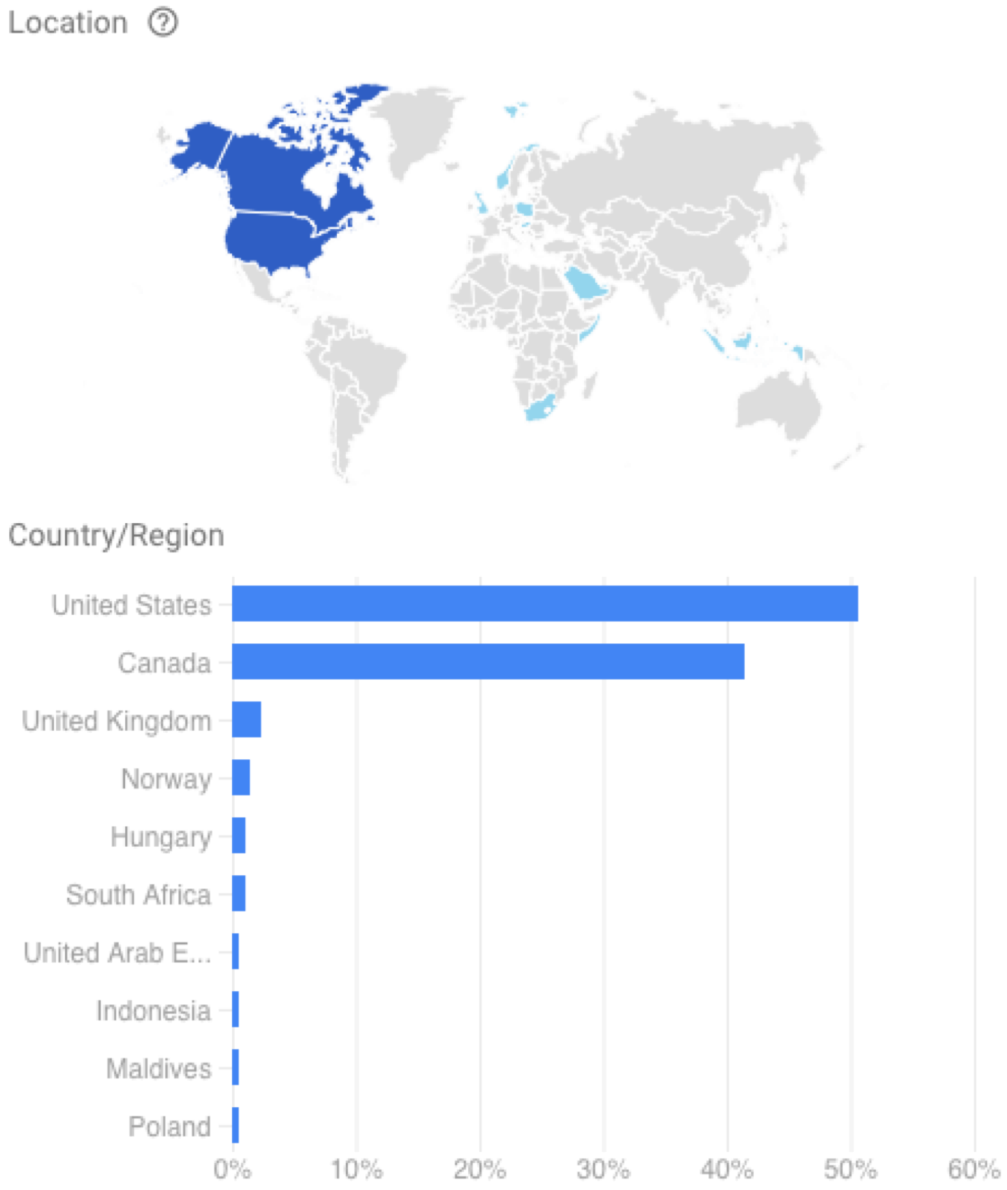
EVENT LOGS

Global Event Logs

Page 1

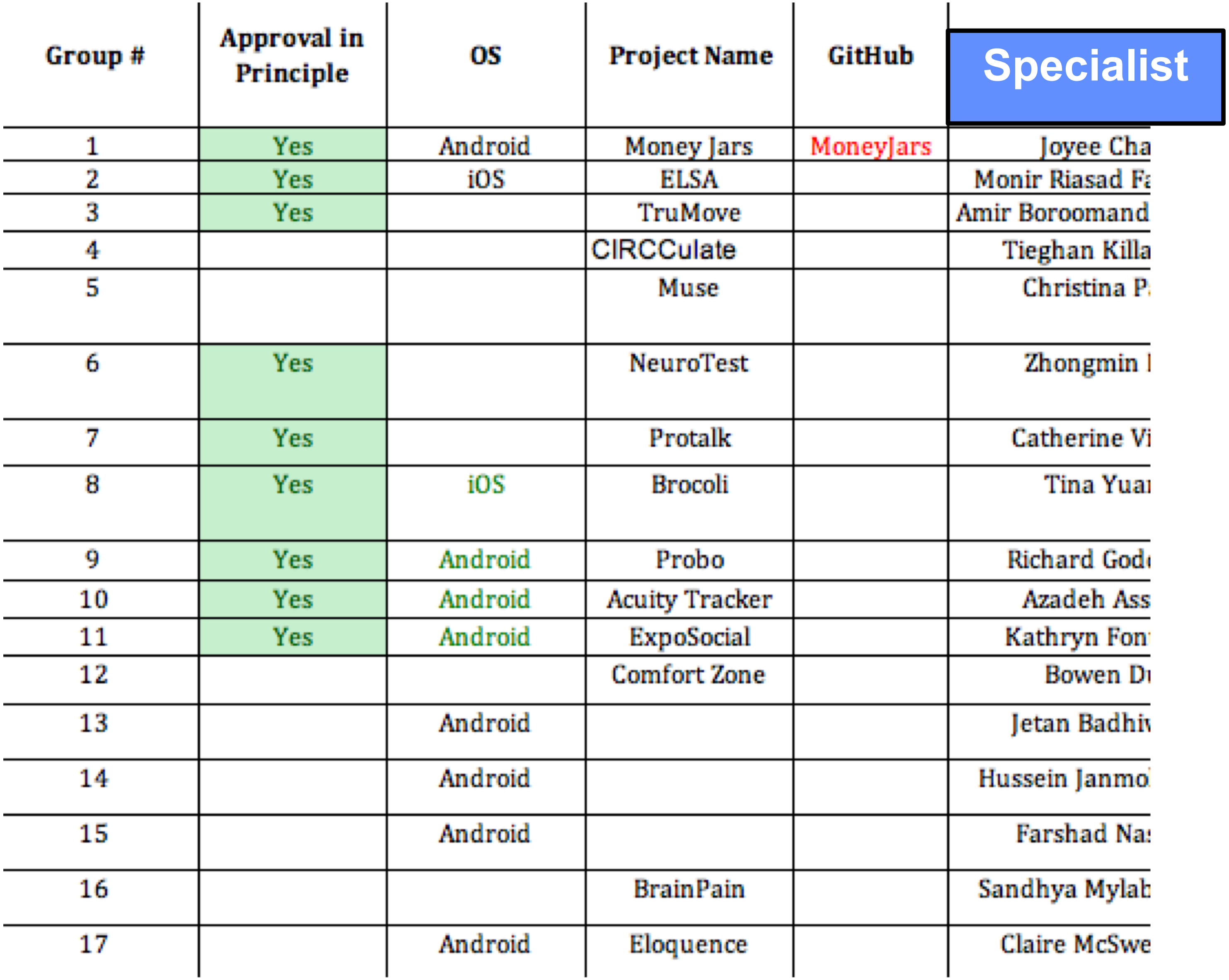
Session Time	Version	Details
⊟ 01/19/14 21:02:57 EST	1.6 (iPhone)	Apple iPhone 4s
⊞ 1) **Teams Made**		
⊟ 01/19/14 12:27:20 EST	1.6 (iPhone)	Apple iPad 2
1) **Adding Players Mode**		
2) **New Player Added**		
⊟ 01/19/14 12:24:27 EST	1.6 (iPhone)	Apple iPad 2
⊞ 1) **Teams Made**		
⊟ 01/18/14 15:54:27 EST	1.6 (iPhone)	Apple iPad 2
⊞ 1) **Teams Made**		
⊟ 01/18/14 15:54:11 EST	1.6 (iPhone)	Apple iPad 2
⊞ 1) **Teams Made**		
⊟ 01/17/14 16:03:22 EST	1.6 (iPhone)	Apple iPhone 4 (GSM)
⊞ 1) **Teams Made**		
⊞ 2) **Teams Made**		
⊟ 01/17/14 15:58:44 EST	1.6 (iPhone)	Apple iPhone 4 (GSM)
1) **Adding Players Mode**		
2) **New Player Added**		
⊞ 3) **Teams Made**		
⊟ 01/16/14 16:14:47 EST	1.6 (iPhone)	Apple iPhone 5 (CDMA)
⊞ 1) **Teams Made**		

Geography


Location ?
Country/Region
United States
Canada
United Kingdom
Norway
Hungary
South Africa
United Arab E...
Indonesia
Maldives
Poland
0%
10%
20%
30%
40%
50%
60%

Topic Discussion Time

Group Names:



Group #	Approval in Principle	OS	Project Name	GitHub	Specialist
1	Yes	Android	Money Jars	MoneyJars	Joyee Cha
2	Yes	iOS	ELSA		Monir Riasad Fa
3	Yes		TruMove		Amir Boroomand
4			CIRCCulate		Tieghan Killa
5			Muse		Christina P
6	Yes		NeuroTest		Zhongmin
7	Yes		Protalk		Catherine Vi
8	Yes	iOS	Brocoli		Tina Yua
9	Yes	Android	Probo		Richard God
10	Yes	Android	Acuity Tracker		Azadeh Ass
11	Yes	Android	ExpoSocial		Kathryn Fon
12			Comfort Zone		Bowen D
13		Android			Jetan Badhi
14		Android			Hussein Janmo
15		Android			Farshad Na
16			BrainPain		Sandhya Mylab
17		Android	Eloquence		Claire McSwe