

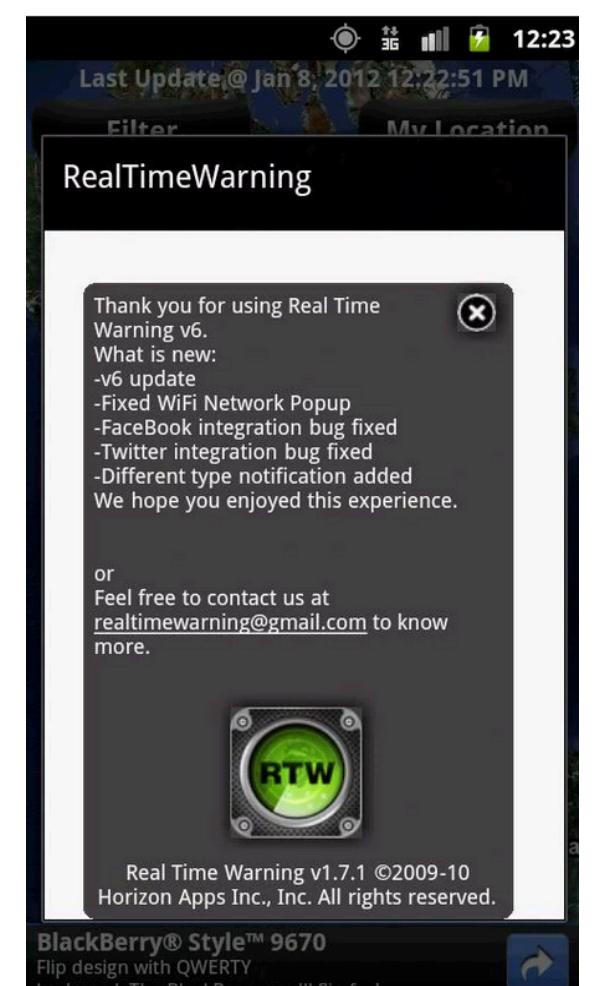
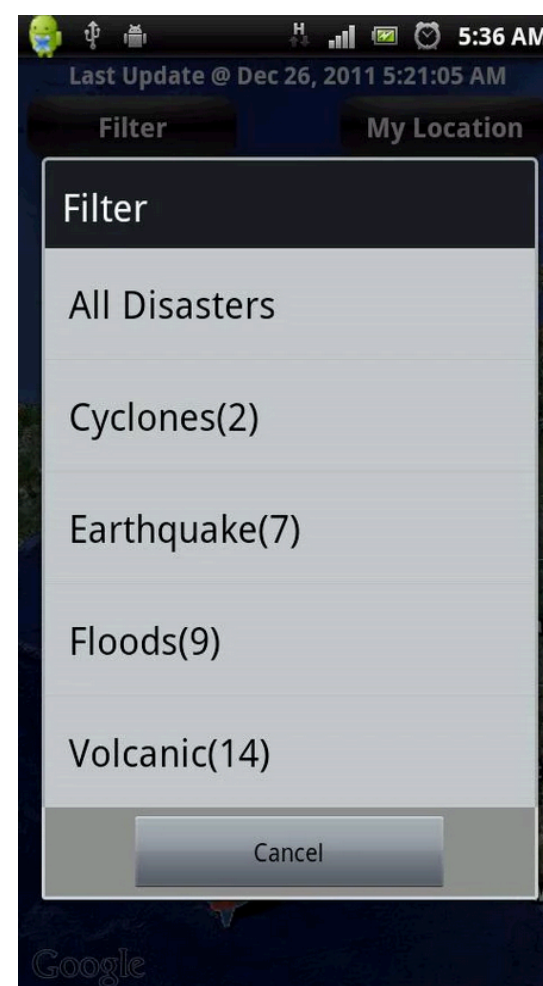
PUNYA Framework:

On the Fly Development & Deployment of Android Apps

Julius Adebayo
CSAIL, MIT

Mobile Ready

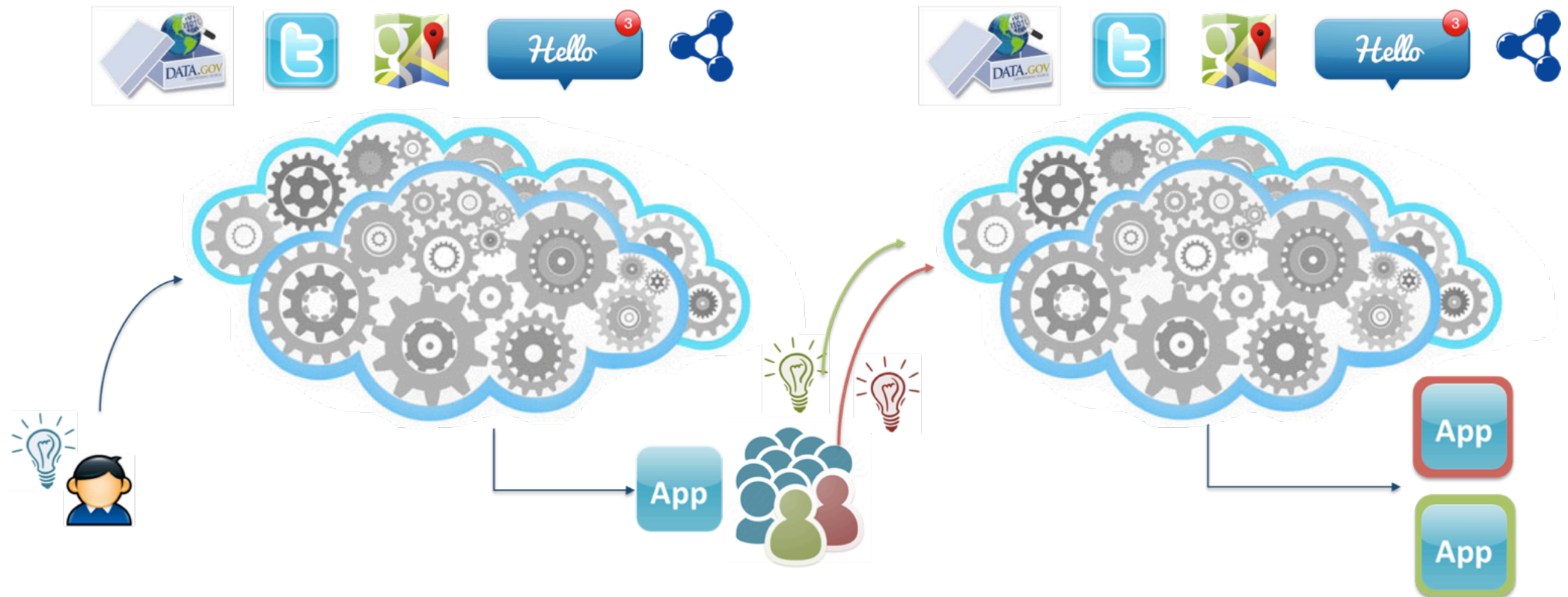
- **Smartphones** are becoming the primary computing platform for people's daily tasks
- Have been found to be useful especially for **disaster management and relief operations**



Current Mobile Development Landscape

- ❖ Detailed requirements for the application
 - Difficult to prototype
- ❖ Budgeting for app development
 - \$75 ~ \$150 per hr for an experienced freelancer, 8~10 weeks to design and complete an app
 - 24k ~ 50k (usd) for complete app
- ❖ Lengthy development time

App Development Platform



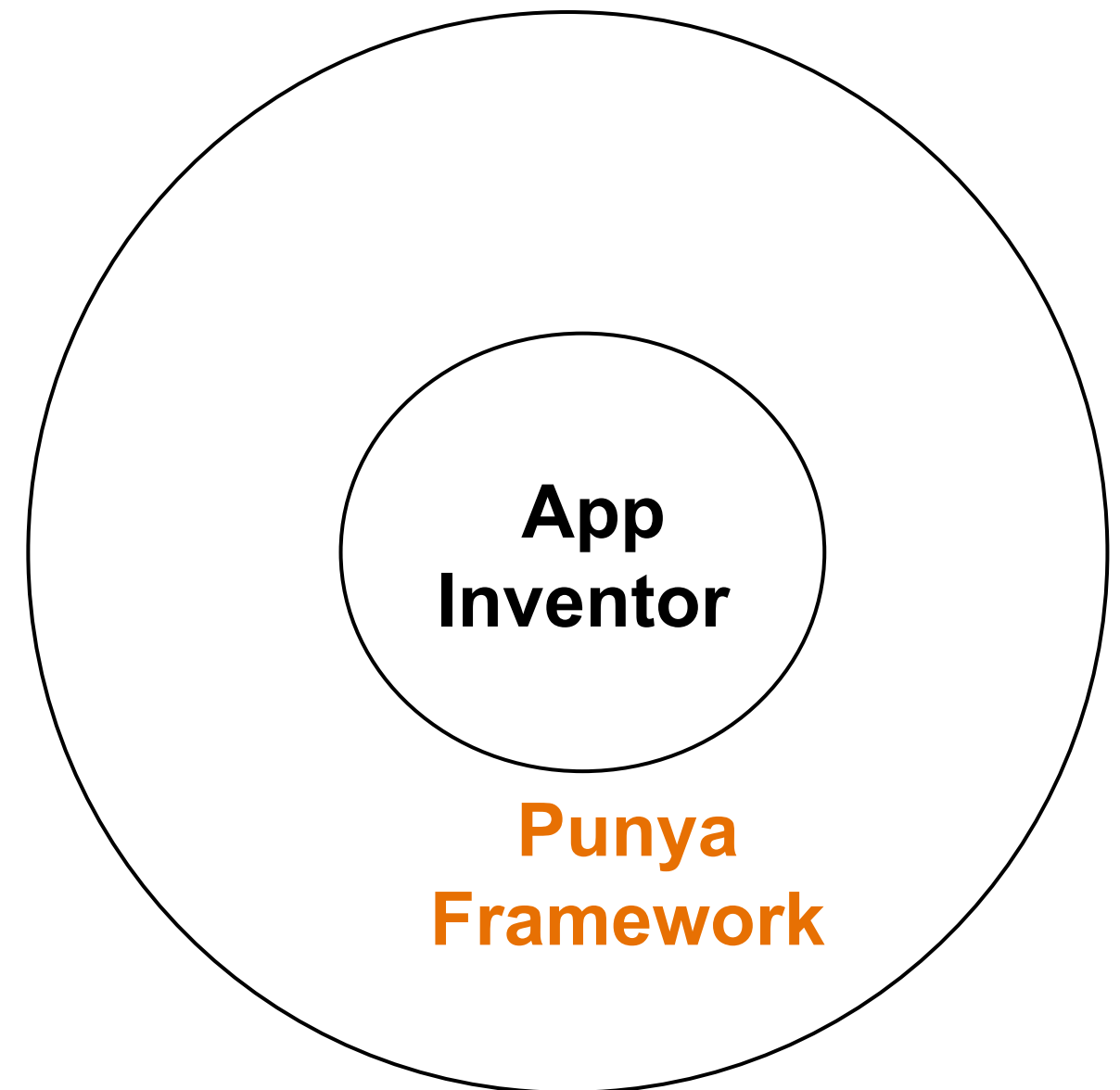
Critical Platform Requirements

- **Requirement 1:** mobile apps are the best way to collect data, communicate, self-organize, etc. during a crisis
 - **Challenge:** Difficult to develop & deploy mobile apps **on the fly**
- **Requirement 2:** Apps need to process large amounts of (heterogeneous) data created & available
 - **Challenge:** Difficult to integrate (heterogeneous) data from different sources
- **Requirement 3:** Humanitarian focused apps require backend/server side processing
 - **Challenge:** Require significant setup time
- **Requirement 4:** Data analytics
 - **Challenge:** Data needs to be processed, analyzed, visualized to provide useful feedback to decision makers.

Our Solution: Punya Framework

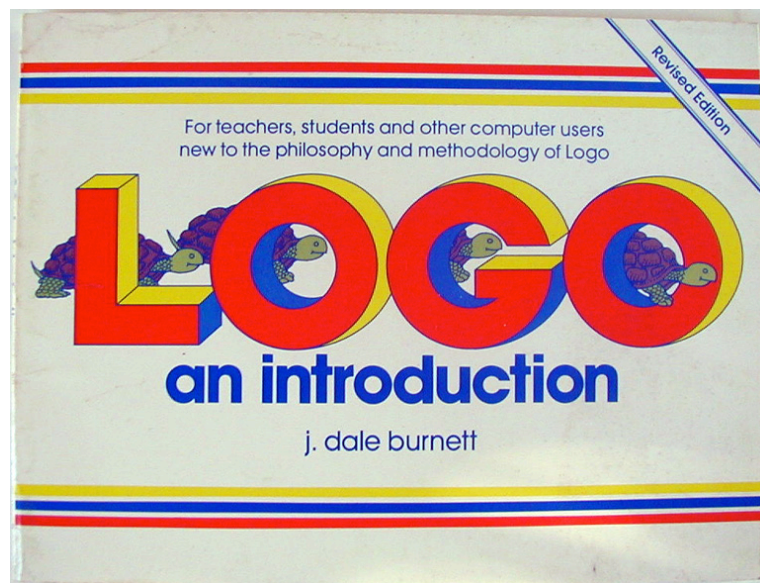
- ***App Inventor***

- Former Google Research Lab Education Project
- Block programming tool for non-programmer to learn about mobile programming
- Designed for students



Our Solution: Punya Framework

Block programming language



SCRATCH



Punya Framework

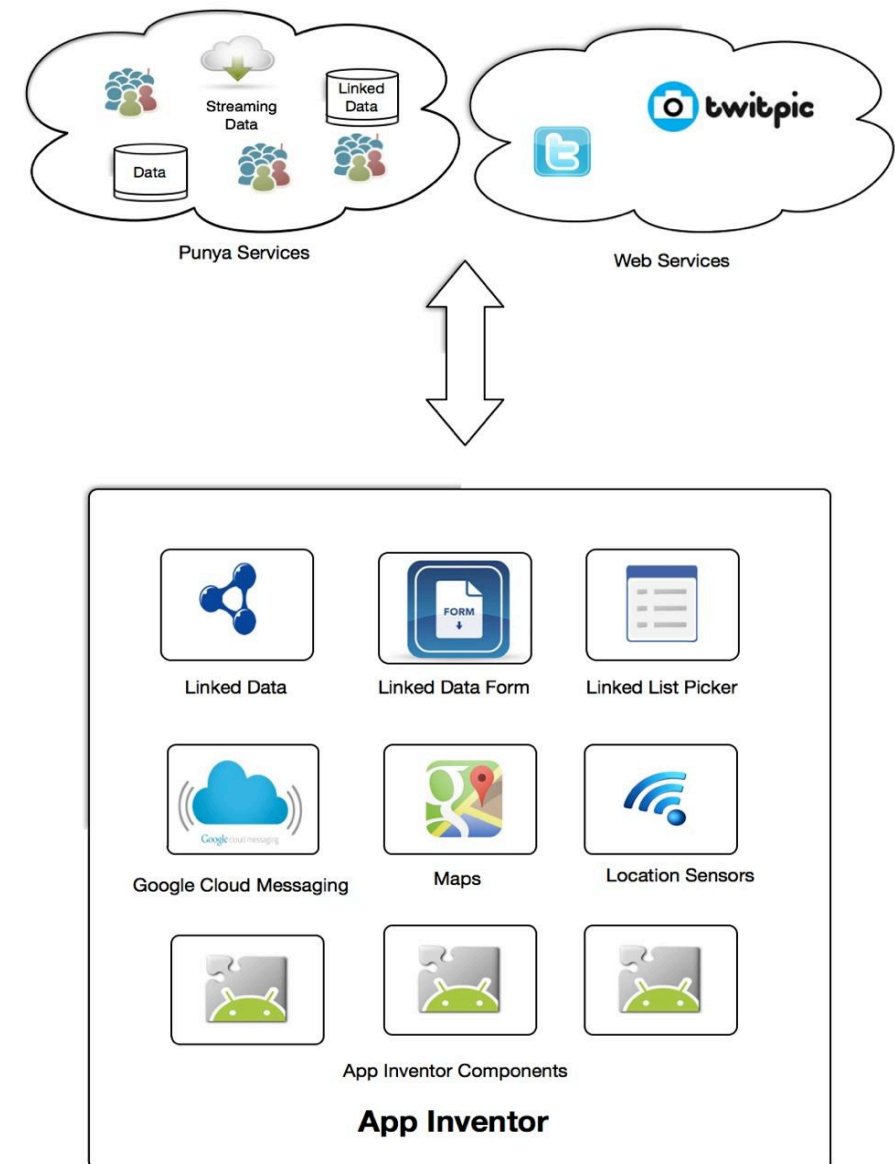
- Vertical integration to provide core functionality for **humanitarian contexts**

- Components**

- **Cloud** messaging component
- Linked **Data** components
- **Sensor** components
- **Map** component

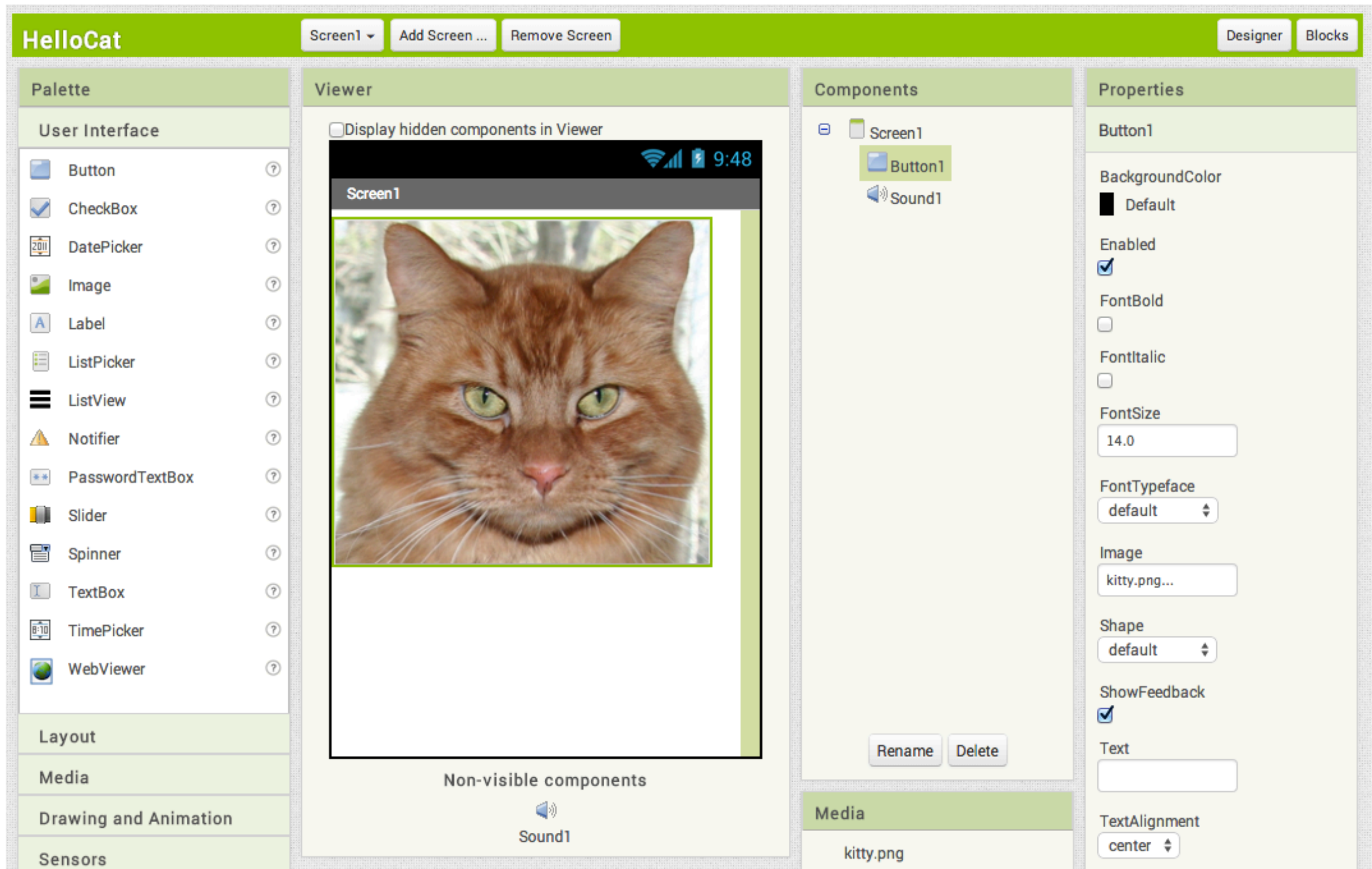
- Services**

- Cloud messaging services
- Linked Data service
- **Stream** data Query



PUNYA Framework

Framework in Practice: Designer View



Framework in Practice: Blocks View

The image shows a software interface with two main panels: "Blocks" on the left and "Viewer" on the right. The "Blocks" panel contains a list of built-in blocks categorized by color: Control (yellow), Logic (green), Math (blue), Text (pink), Lists (light blue), Colors (grey), Variables (orange), and Procedures (purple). Below these categories, there is a section for "Screen1" which includes "Button1" and "Sound1". The "Viewer" panel displays a stack of event listener blocks, each starting with "when Button1" followed by an event name and ".Click", ".GotFocus", ".LongClick", ".LostFocus", or ".TouchDown". The first block, "when Button1 .Click", is highlighted with a red rectangle. The "Screen1" section in the "Blocks" panel is also highlighted with a red rectangle.

Blocks

Built-in

- Control
- Logic
- Math
- Text
- Lists
- Colors
- Variables
- Procedures

Screen1

- Button1
- Sound1

Any component

Viewer

when Button1 .Click

do

when Button1 .GotFocus

do

when Button1 .LongClick

do

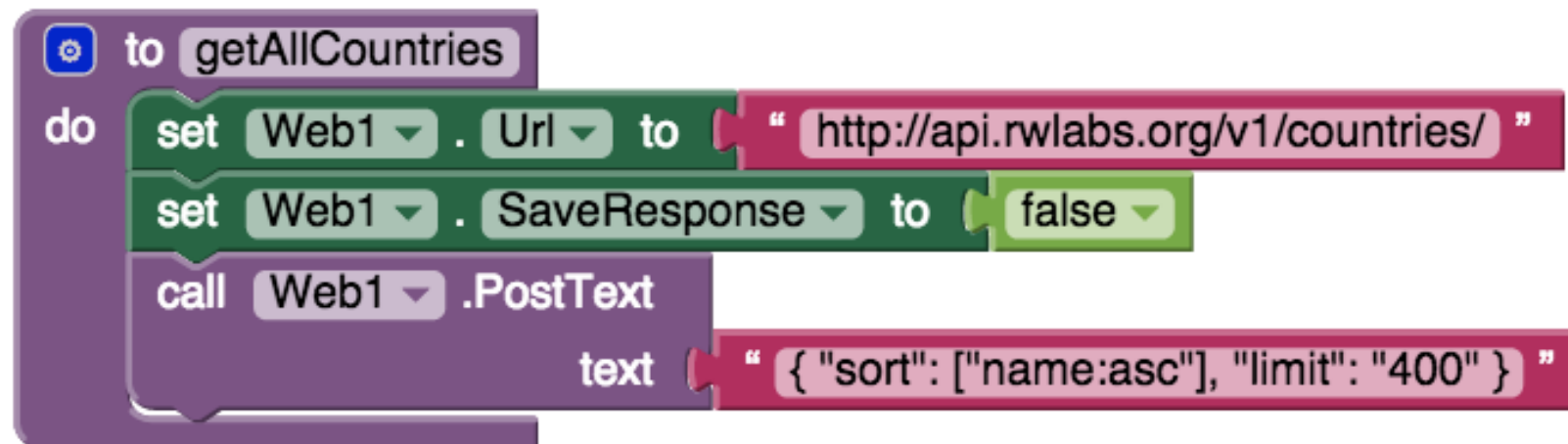
when Button1 .LostFocus

do

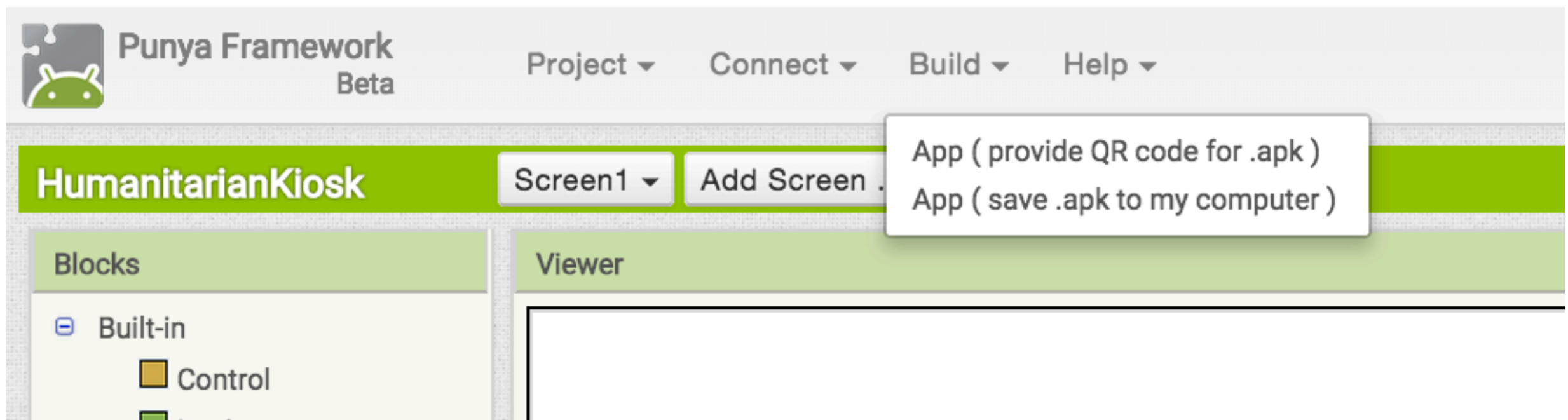
when Button1 .TouchDown

do

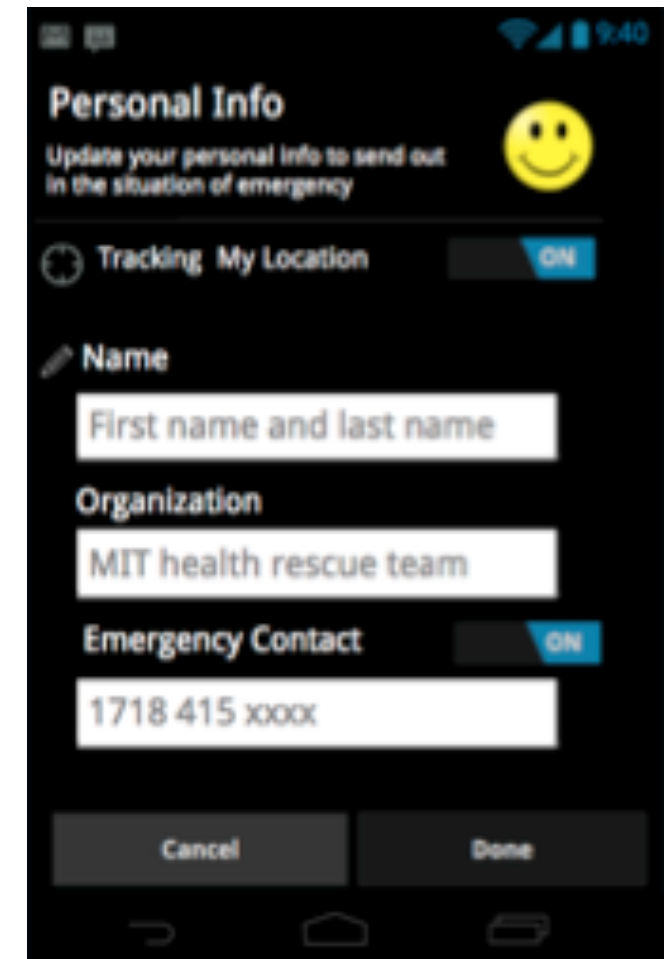
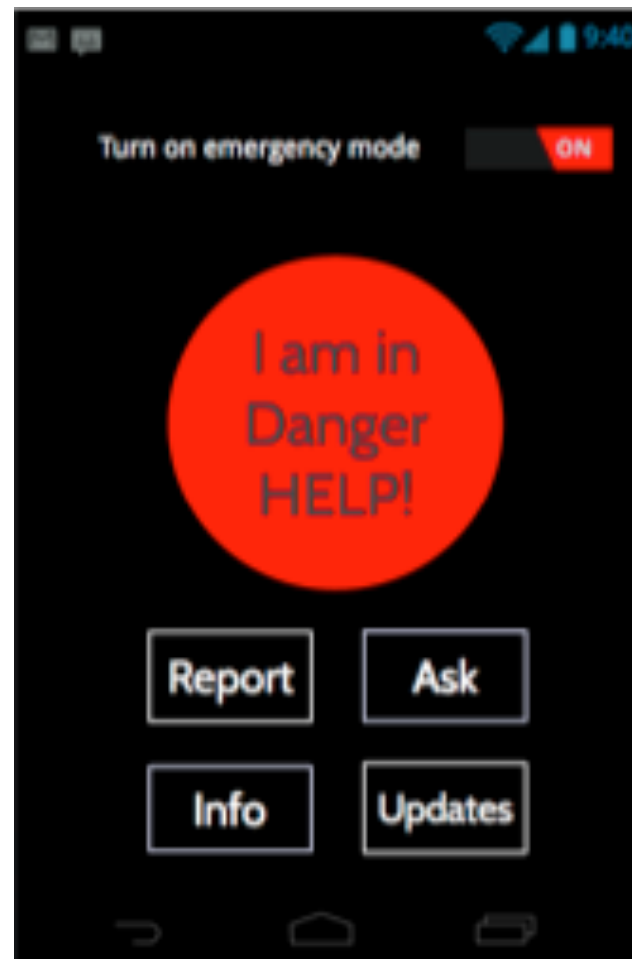
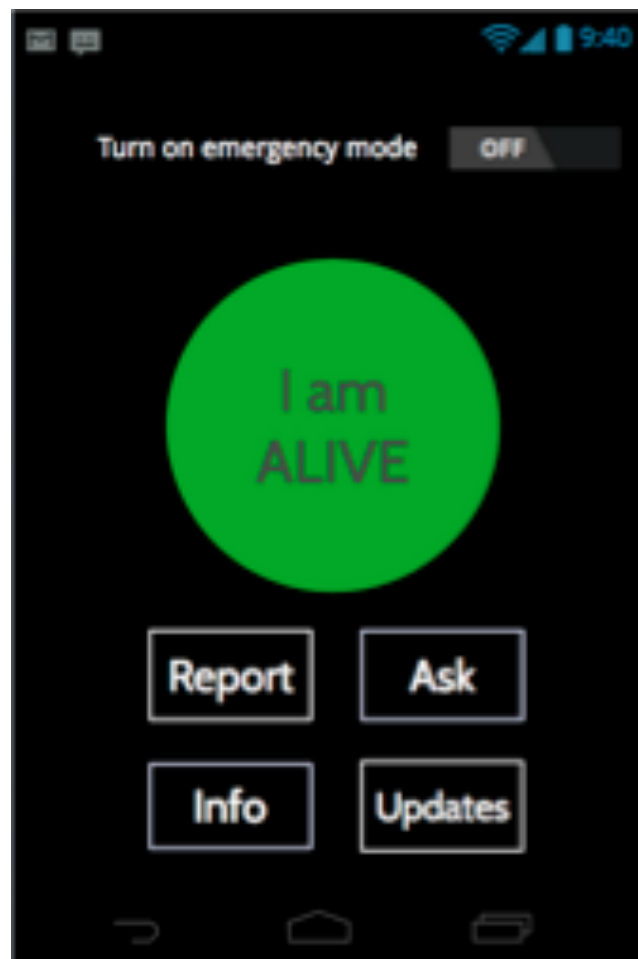
Framework in Practice: Logic



Framework in Practice: Building Your App



Example App: Restoring Family Links



Example App: Restoring Family Links

Report

found ☒ missing ☐

Last seen ? 

Name

Gender male ☒ female ☐

Is the person injured ? ☒

Is the person dead ? ☐

Ask

Results

 Jason Lander
Status: **Missing**
Last seen: Jan 3, 2013
Place: Concord, Massachusetts

 Jason Statham
Status: **Found (require validation)**
Found date: Jan 17, 2013
Place: Mass General Hospital

Update

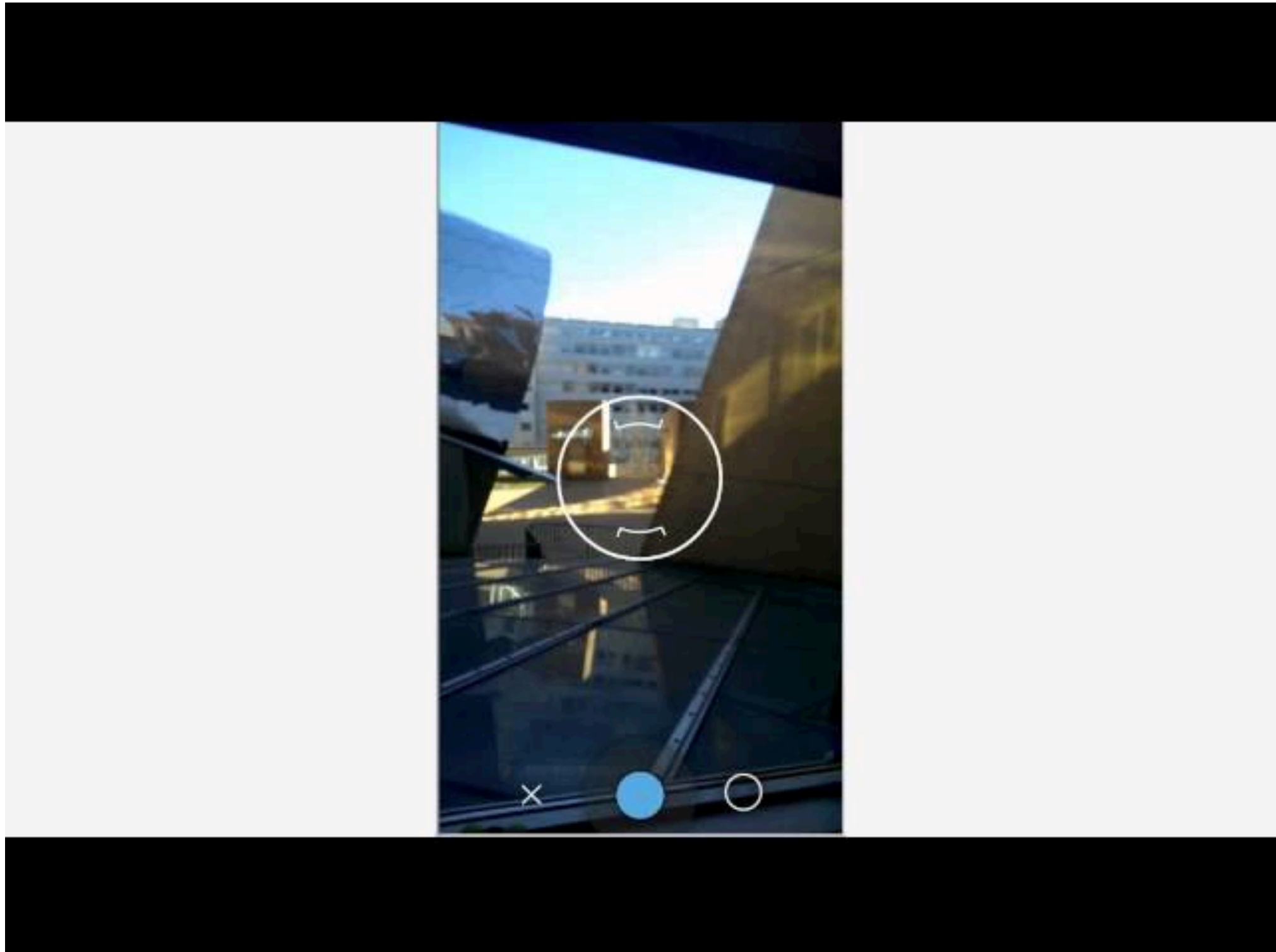
Level: **Warning** 15:34 pm
Armed violence in Downtown Boston. [by certified source](#)

Level: **Warning** 15:33 pm
Someone is shooting in T station [by crowd](#)

Level: **Warning** 15:31 pm
5**E gun shots in South Station [by crowd](#)

Level: **Warning** 15:31 pm
**E gun shots in South Station [by crowd](#)

Example App: WeReport



Example App: WeReport

- **Allows users to submit reports about felled trees, down power lines, flooded roads, etc.**
 - Users are able to take pics, annotate them, and assign an appropriate category (such as flooded roads)
 - Users can also subscribe to receive real time updates of reports about different categories in their area
- WeReport: <https://www.youtube.com/watch?v=3gGJurMHasg>

Current Development

- **Offline tolerance**
 - Web requests are cached and replayed
 - Wifi Direct as another channel
- **Usability of Linked Data**
 - Making Linked Data easier to use and deploy in mobile contexts
- **Template apps**
 - Reuse parts of apps easily
- **Privacy**
 - Allow data to be collected, stored, and used in a privacy aware manner

Trying it out

Try it:
punya.mit.edu

Contact us:
punya-info@csail.mit.edu

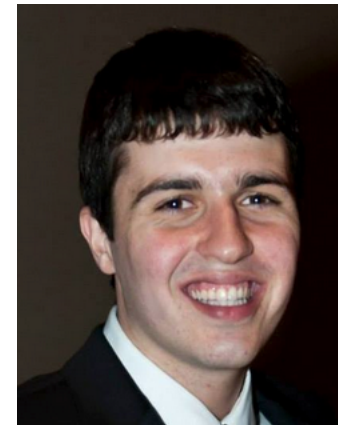
Project Team



WeiHua Li
 MEng @ MIT



Julius Adebayo
 SM @ MIT



Eduardo Leon
 Undergrad @ MIT



Anubhav Jain
 Undergrad @ MIT



Andrew McKinney
 Tech Lead @ MIT App Inventor



Carlos Castillo
 QCRI PI



Patrick Meier
 QCRI PI



Lalana Kagal
 MIT PI