Use Case Revisited

- Electronic circuits are **potentially dangerous** and **lack accessibility**
- **Benefits to drawing circuits** on paper
  - Good to connect symbols to components
  - Kids like to draw!
- Solution: build a system that **takes a picture** of a drawn circuit, **simulates circuit**, and **renders annotated schematic**
  - Learning and verification tool
- Target users: **middle school students**

Use Case Requirements Revisited

- Usability
  - Accessibility: application is **free of cost**
  - **80% approval rate** from test group
- Individual Component Detection Accuracy
  - **90% detection accuracy** from unit tests
- Combined Component Detection Accuracy
  - Display **correct circuit 90%** of the time
- Simulator Accuracy
  - **100% correctness** on analyzing given circuit

Design Requirements

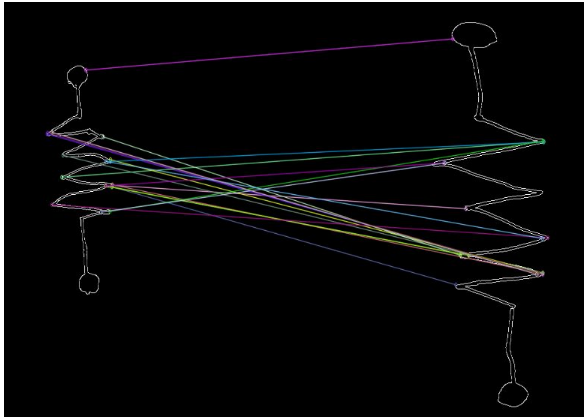
- App for **iOS Devices**
- Components must be **at least 350x200 pixels**
    - Can fit 8 components onto standard **1080p image** while maintaining quality
- iOS Version >= 8.0
    - iPhone 6 and above

Solution - Mobile Application

- Swift application for **iOS devices**
- Objective-C++ Backend
  - OpenCV
  - Custom Circuit Simulator
- **Works offline**, no need for internet connection
- Easy to use, no need to upload image to computer

Solution - Circuit Detection

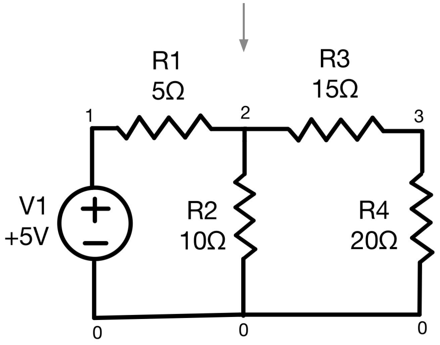
- **Detect nodes** at end of components
    - Hough Circles
- Find components and wires between nodes
- Run ORB on components to detect features
- Use **brute force matching** to match features
- Display **five circuits** with highest confidence
    - Reduces error in circuit detection for the end user
- Users will no longer write component values
    - Enter in app to reduce error from CV



Solution - Circuit Simulation

- Receive **netlist** generated by user inputting values after confirming circuit
- Parse netlist into graph data structure
- Run simulation by performing **nodal analysis**
- Steady state **DC analysis** for MVP
  - Voltage/current sources, resistors, lightbulbs, switches, LEDs

```
R1 1 2 5
R2 2 0 10
R3 2 3 15
R4 3 0 20
V1 1 0 5
```



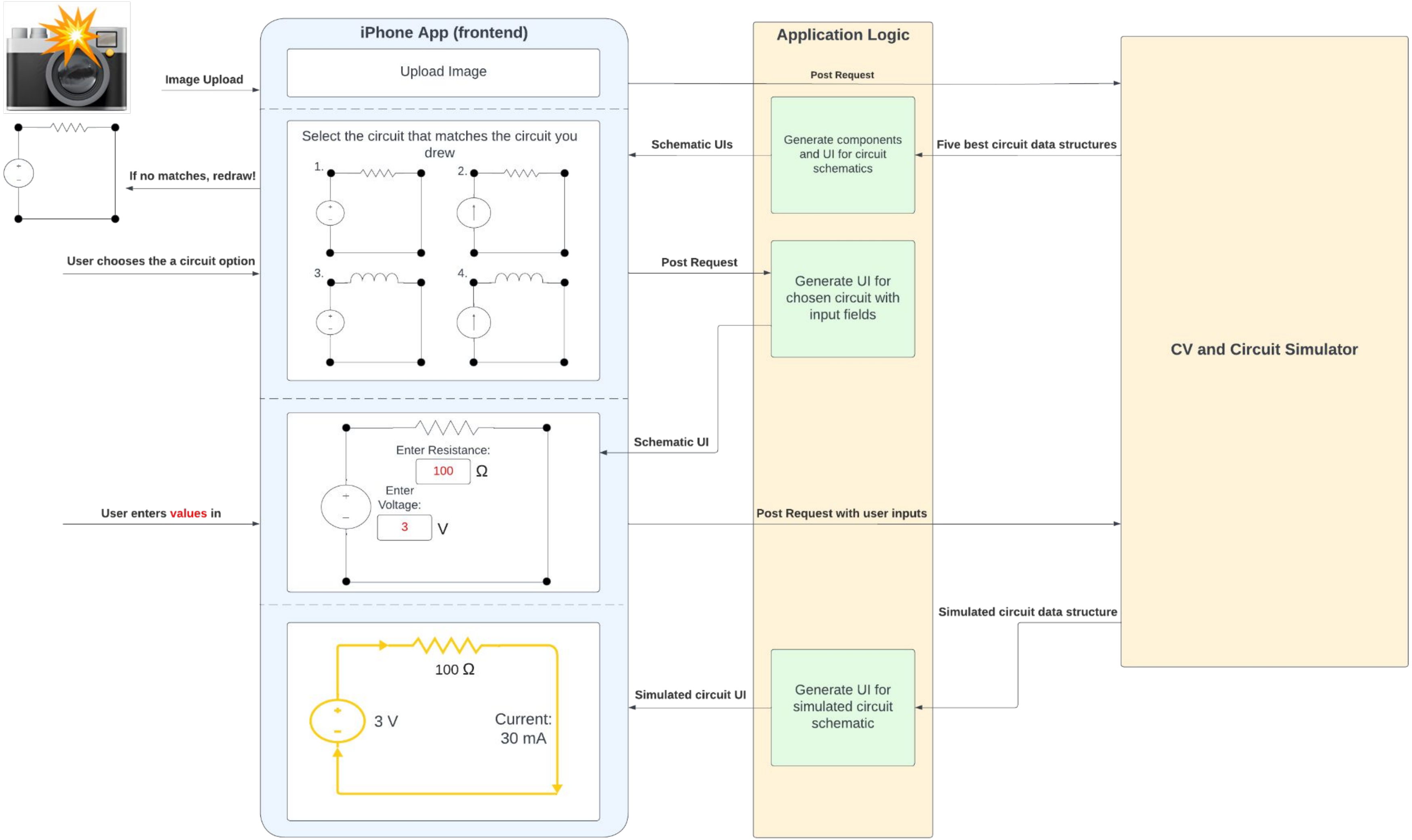


Image Upload

If no matches, redraw!

User chooses the a circuit option

User enters **values** in

iPhone App (frontend)

Upload Image

Select the circuit that matches the circuit you drew

1.

2.

3.

4.

Enter Resistance:

100 Ω

Enter Voltage:

3 V

100 Ω

3 V

Current:
30 mA

Application Logic

Post Request

Schematic UIs

Generate components and UI for circuit schematics

Five best circuit data structures

Post Request

Generate UI for chosen circuit with input fields

Schematic UI

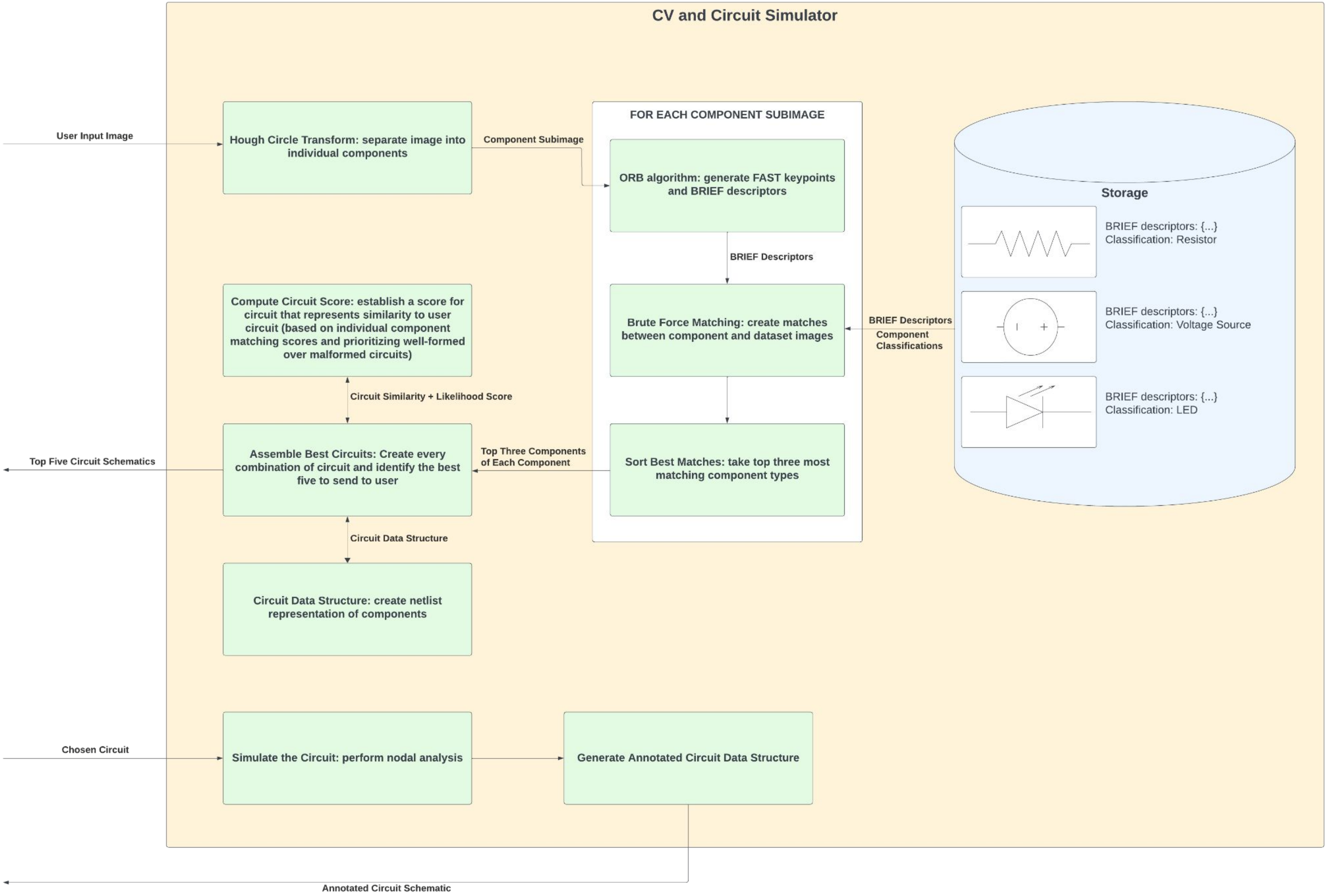
Post Request with user inputs

Simulated circuit data structure

Simulated circuit UI

Generate UI for simulated circuit schematic

CV and Circuit Simulator



CV and Circuit Simulator

User Input Image

Hough Circle Transform: separate image into individual components

Component Subimage

FOR EACH COMPONENT SUBIMAGE

ORB algorithm: generate FAST keypoints and BRIEF descriptors

BRIEF Descriptors

Brute Force Matching: create matches between component and dataset images

Sort Best Matches: take top three most matching component types

Top Three Components of Each Component

Storage

BRIEF descriptors: {...}
Classification: Resistor

BRIEF descriptors: {...}
Classification: Voltage Source

BRIEF descriptors: {...}
Classification: LED

BRIEF Descriptors

Component Classifications

Compute Circuit Score: establish a score for circuit that represents similarity to user circuit (based on individual component matching scores and prioritizing well-formed over malformed circuits)

Circuit Similarity + Likelihood Score

Assemble Best Circuits: Create every combination of circuit and identify the best five to send to user

Top Five Circuit Schematics

Circuit Data Structure

Circuit Data Structure: create netlist representation of components

Chosen Circuit

Simulate the Circuit: perform nodal analysis

Generate Annotated Circuit Data Structure

Annotated Circuit Schematic

Testing and Verification - Frontend & Simulator

- Frontend Testing
  - Test group of seven **12-14 year olds**
  - Survey UI/UX satisfaction on scales of 1-10
- Circuit Simulation Testing
  - **Write script** to generate SPICE netlists
  - Use existing circuit simulation tools to test against
    - LTspice
  - Simulations should **match every time**

Testing and Verification - Computer Vision

- Individual Component Detection
    - Test group will draw **six drawings of each component type**
        - Will rotate components to **test all orientations**
    - **90 percent** testing accuracy
- Full Circuit Detection
    - Test group will draw **three circuits each**
    - We will draw many circuits for additional validation
    - Correct circuit should appear in top 5 circuits 90% of the time
- Risk Mitigation
    - Increase number of displayed circuit choices

TASK TITLE	TASK OWNER	09/17							09/24							10/01							10/08							10/15							10/22							10/29							11/05							11/12							11/19							11/26						
		S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S	S	M	T	W	R	F	S
Logistics																																			⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Design presentation	Everyone								🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨																			⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Design review/report	Everyone																				🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨							⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Final presentation	Everyone																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛	🟨	🟨	🟨	🟨	🟨	🟨	🟨
iOS Application																																			⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Create dummy app in XCode and load onto phone	Jaden															🟦	🟦	🟦	🟦																⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Add camera/upload image feature to app	Jaden																			🟦	🟦	🟦	🟦	🟦											⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Create frontend models of components	Jaden																								🟦	🟦	🟦	🟦	🟦						⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Construct circuit from netlist	Jaden																													🟦	🟦	🟦	🟦	🟦	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	🟦	🟦																							⬛	⬛	⬛	⬛							
Add user value input and construct circuit from netlist with labelled simulation values	Jaden																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛			🟦	🟦	🟦	🟦	🟦	🟦	🟦																		⬛	⬛	⬛	⬛							
Computer Vision																																			⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Research image detection	Everyone		🟨	🟨	🟨	🟨	🟨	🟨																											⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Identify resistor and wire individually	Stephen								🟥	🟥	🟥	🟥	🟥	🟥	🟥																				⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Correctly identify resistor and wire in a single circuit	Stephen															🟥	🟥	🟥	🟥	🟥	🟥	🟥													⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Identify voltage and current source individually	Stephen																						🟥	🟥	🟥	🟥	🟥	🟥	🟥						⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Correctly identify voltage and current source in a single circuit	Stephen																													🟥	🟥	🟥	🟥	🟥	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Identify diodes and switches individually	Stephen																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	🟥	🟥	🟥	🟥	🟥																				⬛	⬛	⬛	⬛							
Correctly identify diodes and switches in a single circuit	Stephen																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛						🟥	🟥	🟥	🟥	🟥																	⬛	⬛	⬛	⬛							
Correctly identify all other components in a single circuit	Stephen																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛											🟥	🟥	🟥	🟥	🟥												⬛	⬛	⬛	⬛							
Train and test model with circuit drawings	Everyone																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																🟨	🟨	🟨	🟨	🟨	🟨	🟨			⬛	⬛	⬛	⬛							
Circuit simulator																																			⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Create circuit data structure to send to/from website and CV algorithm	Everyone		🟨	🟨	🟨	🟨	🟨	🟨	🟨																										⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Create internal component object representation of resistor and voltage source	Devan									🟪	🟪	🟪	🟪	🟪	🟪	🟪	🟪																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Create node component to perform analysis	Devan																	🟪	🟪	🟪	🟪	🟪	🟪	🟪											⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Connect components and nodes together in graph	Devan																								🟪	🟪	🟪	🟪	🟪	🟪	🟪				⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Detect valid/invalid circuit	Devan																															🟪	🟪	🟪	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Analyse circuits with voltage source(s) + resistor(s)	Devan																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	🟪	🟪	🟪	🟪	🟪	🟪	🟪																		⬛	⬛	⬛	⬛							
Create diode, switch, current source components	Devan																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛								🟪	🟪	🟪	🟪	🟪													⬛	⬛	⬛	⬛							
Analyse circuit with all components	Devan																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛													🟪	🟪	🟪	🟪	🟪										⬛	⬛	⬛	⬛							
Integration/Final Testing																																			⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																									⬛	⬛	⬛	⬛							
Create database	Jaden																																🟦	🟦	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	🟦	🟦	🟦	🟦																					⬛	⬛	⬛	⬛							
Create wrappers to interface backend with frontend	Jaden																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛					🟦	🟦	🟦																		⬛	⬛	⬛	⬛							
Test pipeline from CV -> simulator	Jaden																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛								🟦	🟦	🟦	🟦														⬛	⬛	⬛	⬛							
Test pipeline from simulator -> frontend	Jaden																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛												🟦	🟦	🟦	🟦												⬛	⬛	⬛	⬛							
Test full pipeline	Jaden																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																🟦	🟦	🟦	🟦	🟦	🟦				⬛	⬛	⬛	⬛							
Slack	Everyone																																		⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛	⬛																						🟨	🟨	🟨	⬛	⬛	⬛	⬛	🟨	🟨	🟨	🟨	🟨	🟨	🟨

Conclusion

- Use-case: **lack of accessibility** for learning basic electronic circuits
- MVP
  - iOS app to capture images and display simulated circuit
  - CV algorithm that can detect components and circuits (8 components maximum)
  - Fully functional DC circuit simulator with limited components