Pocket Microscope Slide Scanner

Team Apollo: Michael Camerino, Jeff Crawford, Archana Dahal, Isaac Riggs, Nico Yensen

Presentation Outline

- Review from Last meeting
  - Product Requirement
  - Design Comparison
  - Budget
- Progress Report
  - Prototyping
  - Preliminary data collection
  - Software Development
- Question and Discussion

Review from the Last Meeting

Scope of this project

Pocket Slide Microscope → Machine Learning Grid → Cancer Diagnosis and Treatment

Children's Cancer Therapy Development Institute

Project Requirements

Cost Requirement

- Device will cost < $1000

User Interface Requirements

- Minimal buttons
- Interface smartphone via Apollo app support
- Live view of slide
- Coarse and fine adjustments to adjust field

Storage and Battery Requirements

- MicroSD card support
- Utilization on phone storage
- Connectivity with online cloud storage
- Minimum of 8000 acquisitions per charge cycle

Optics and Camera Requirements

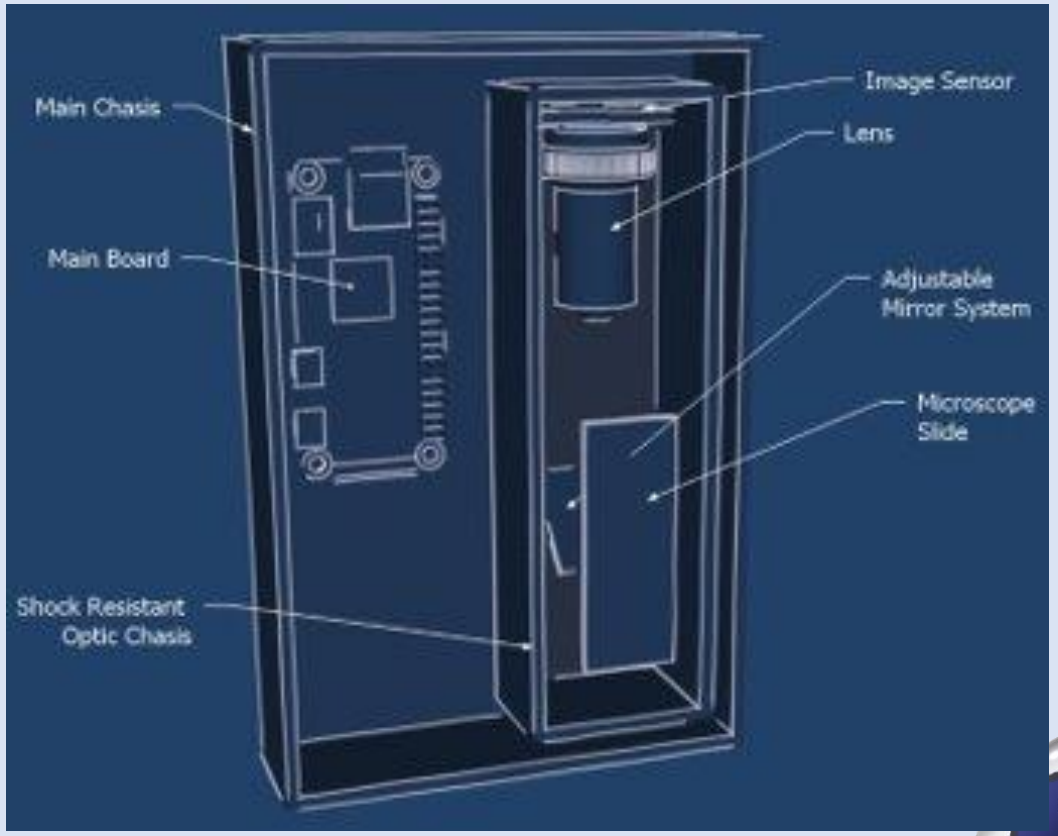
- 40X effective zoom
- 0.5 µm/pixel resolution
- large format imaging capability (up to 1cm2)
- LED light source (5600K)
- Auto white-balance (optional)
- Auto-focus

Networking Requirements

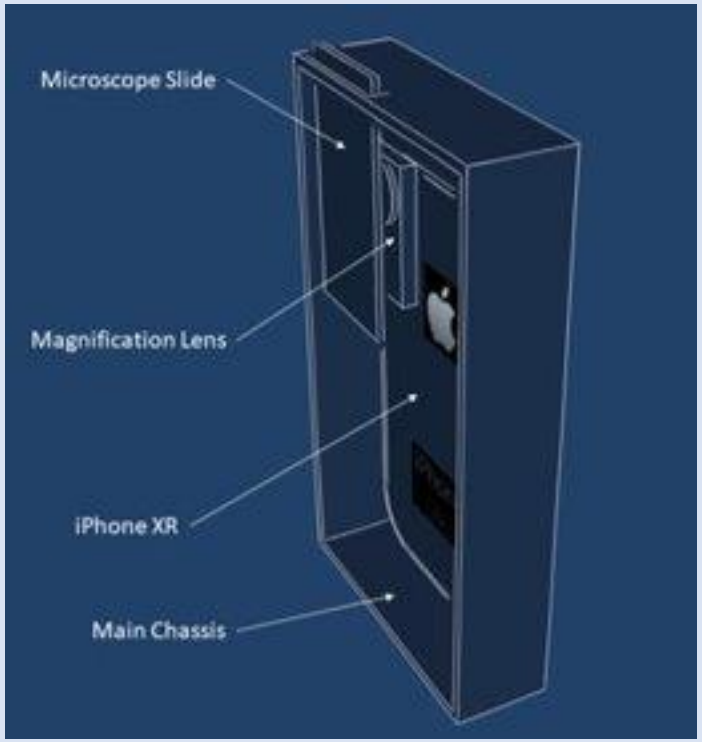
- Wi-Fi (802.11 b/g/n compliant)
- Bluetooth 4.1 compliant
- 4G LTE when paired with smartphone

Design A: Standalone Device

- Utilizes a Raspberry Pi 4 to process and distribute image to phone
- RPi4 Compatible CMOS image sensor
- Bluetooth connectivity to phone
- Phone app for device control and image stitching



Design B: Device Integrated as a Phone Case



- Magnification Lens is available as a consumer product
- Utilizes a phone for image capture, processing, and stitching
- Requires a phone app for device control and stitching

Design Comparison

Concept A

Pros:

- Superior image quality by a factor of 6
- Longer product life
- Less Cost

Cons:

- Larger form factor
- More software development

Concept B

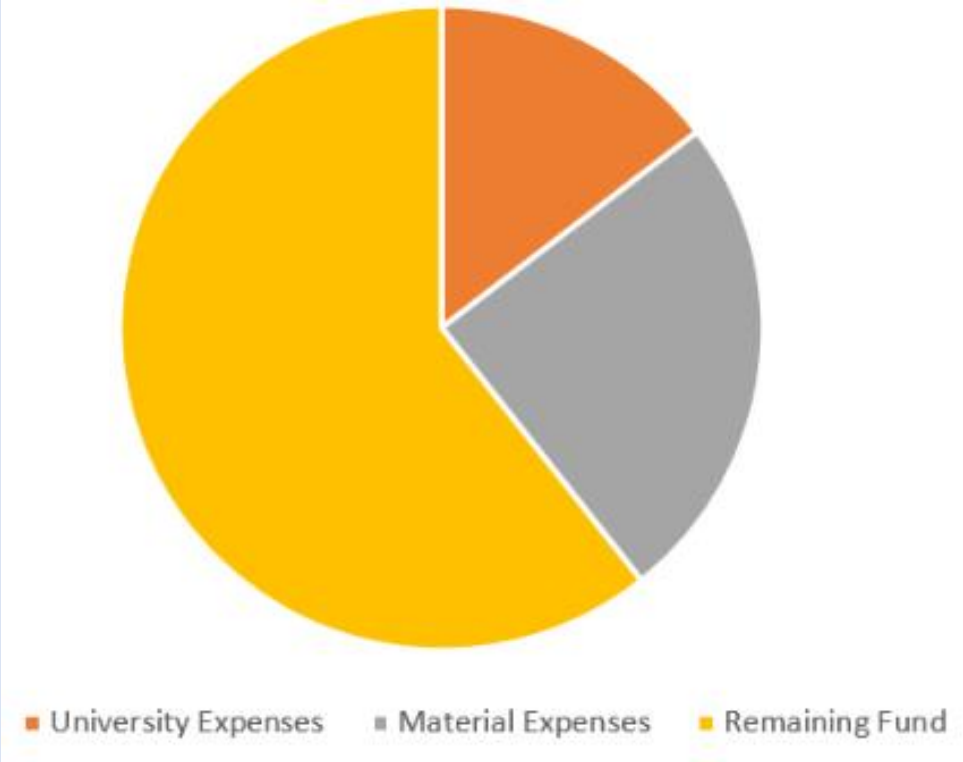
Pros:

- Smaller form factor
- Simpler design
- Easier to implement
- Less peripherals

Cons:

- Lesser image quality
- More cost

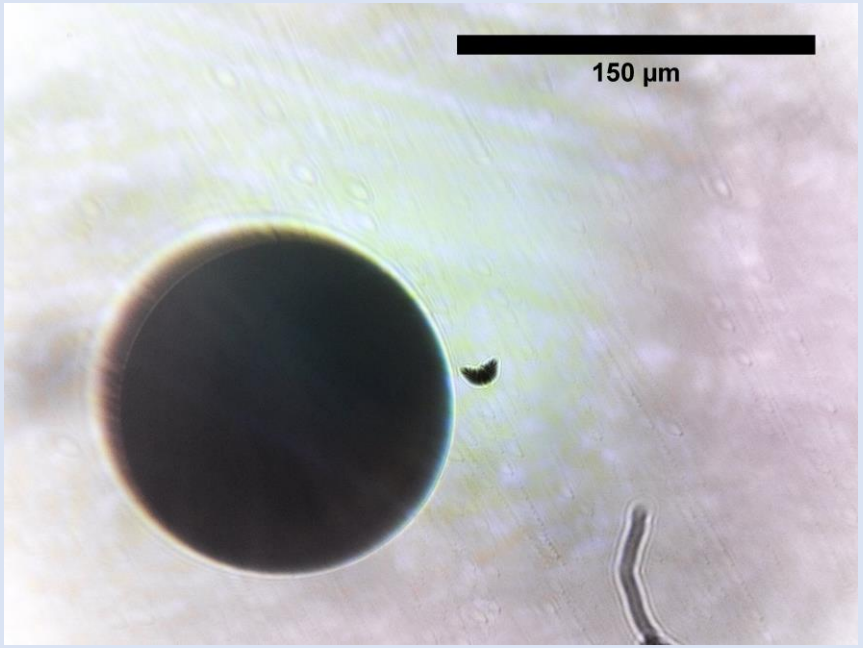
Project Budget

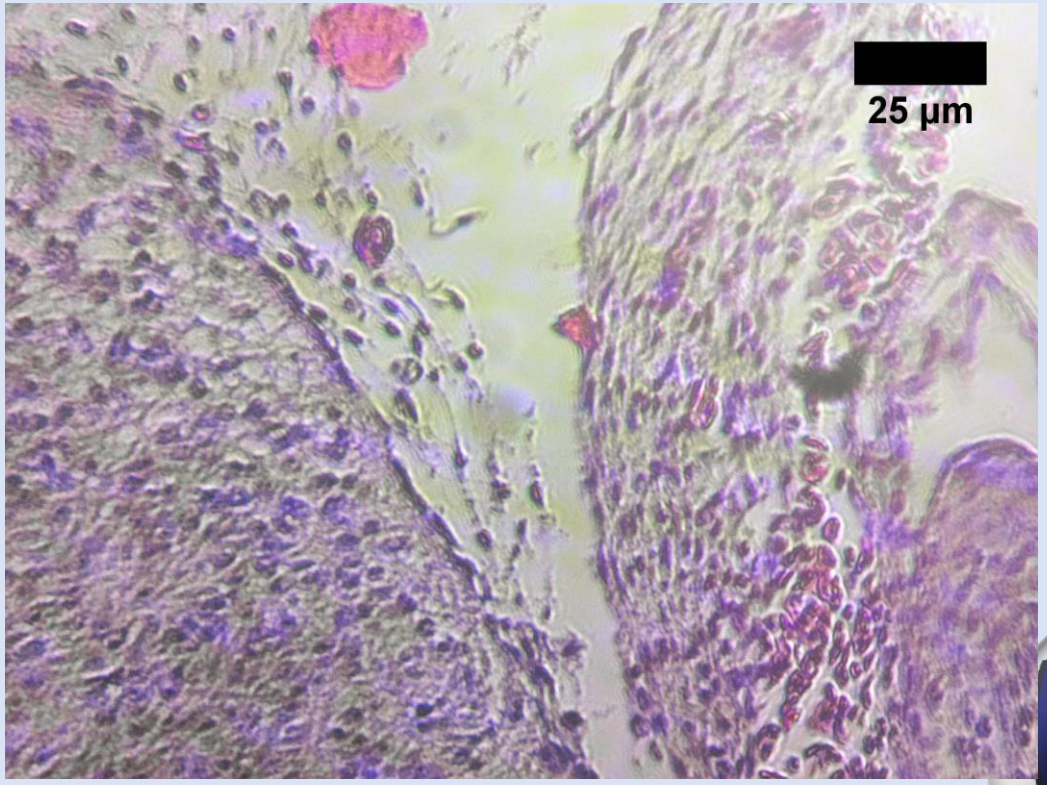


Types of Expenses	Amount
University Expenses	$730
Material and Parts Purchased	$1,236.63
Remaining Fund	$3,033.37
Total Budget	**$5,000**

Progress Report: Design and Software Development

Design Concept A





Design Concept A: Prototyping

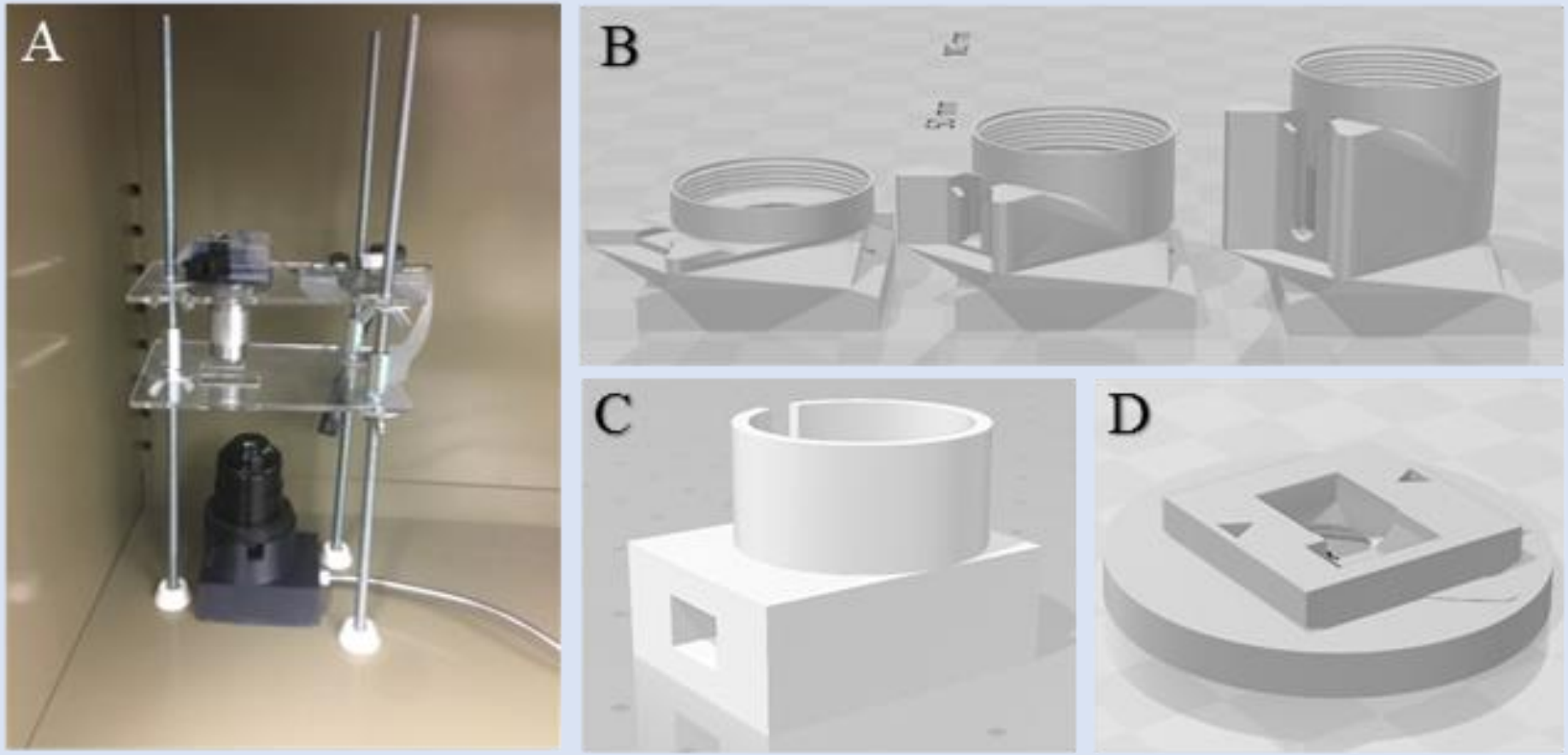


Figure 6 Design Concept A, Rev 0 prototyping: A) Prototype build. B) Series of custom fitted RMS threaded microscope objective/PiCam V2.0 sensor adapters. C) Condenser light source adapter D) PiCam V2.0 fitting.

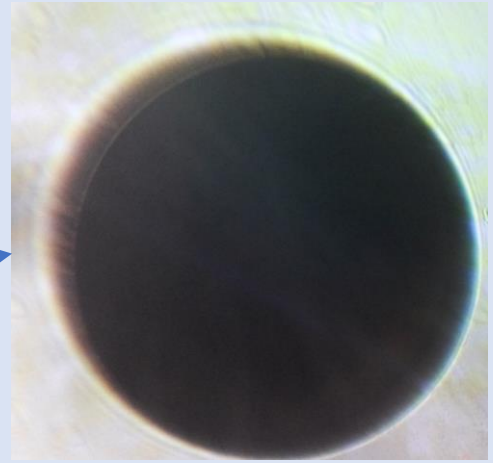
Design Concept A

Current Challenges

- Chromatic aberrations

Solution: Prototype new stage that doesn't have any material between the light and the sample

- Field of focus shift due to high magnification and shallow depth of field

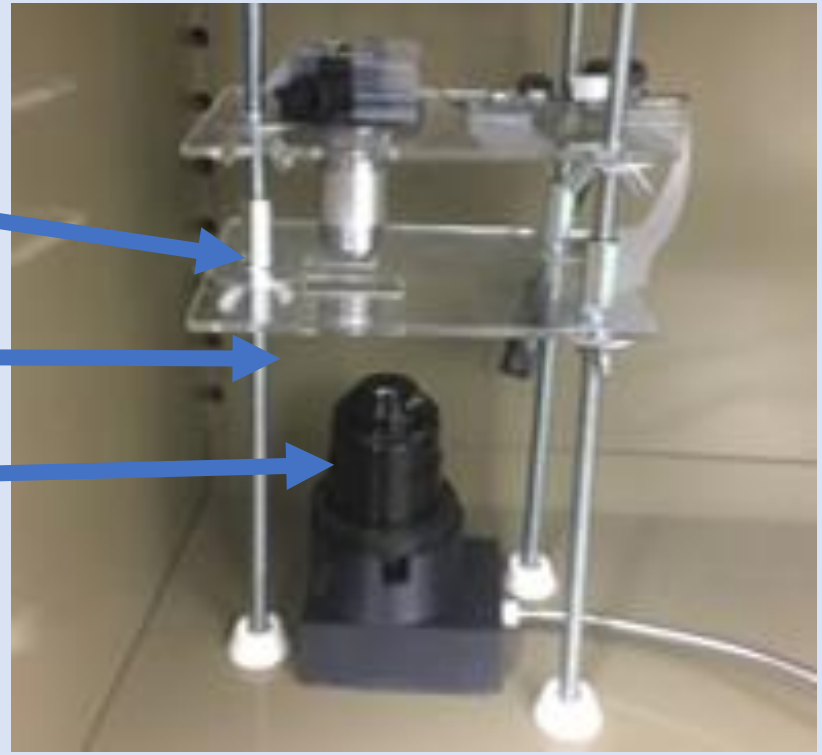


Solution: Better leveling system with one constraint for control instead of 3

Design Concept A

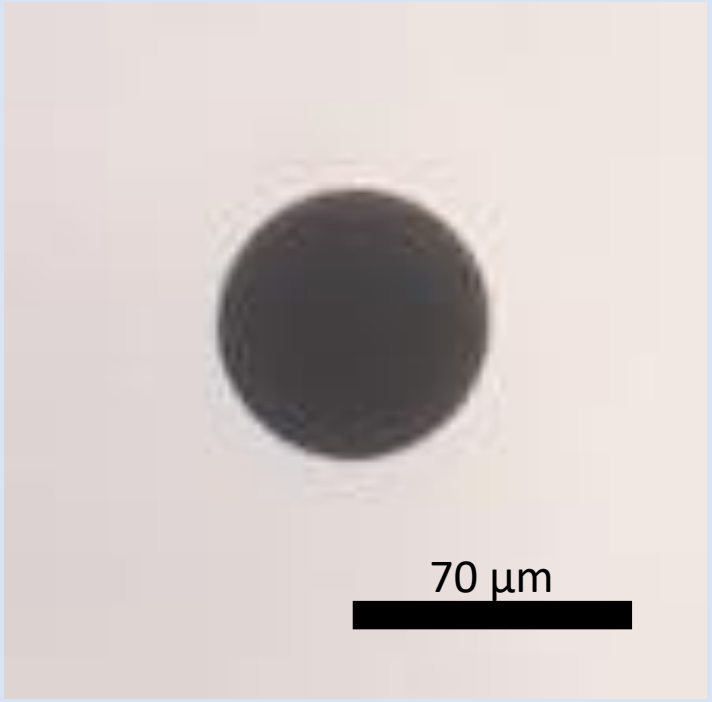
Future work

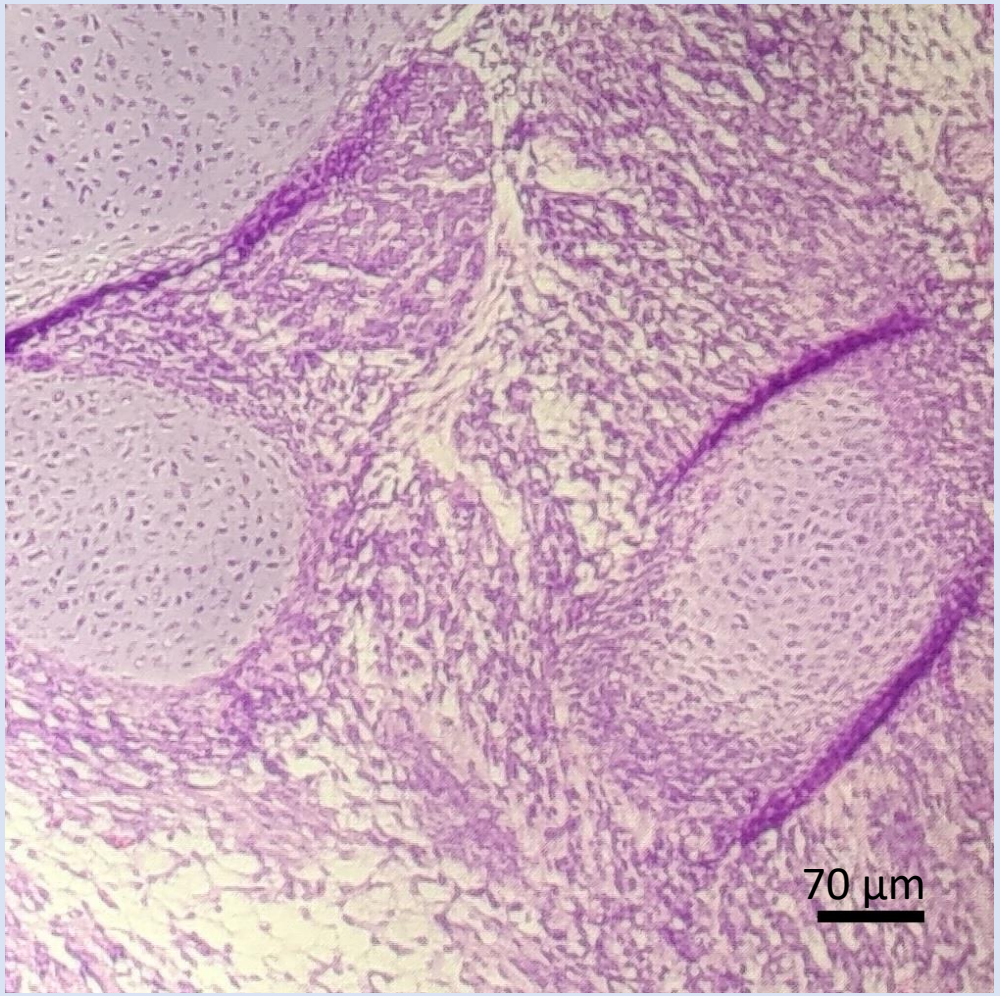
- Prototype new stage with better leveling system
- Minimize material between light and sample
- Reduce the size of condenser
- Consider using 20X objective instead of 40X



Design Concept B
Prototype Progress

Design Concept B
Prototype Progress

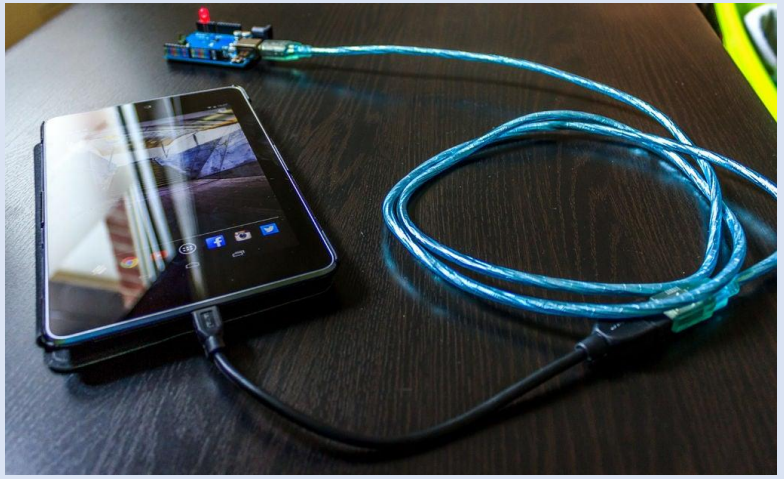




Design Concept B

Current Challenges

- Vignette
- Magnification
- Powering peripherals



Design Concept B

Future Work

- Prototyping with 2X magnification
- Should we choose this design:
  - Stage Design and Prototyping
  - Enclosure Design and 3D printing
  - Lighting Integration

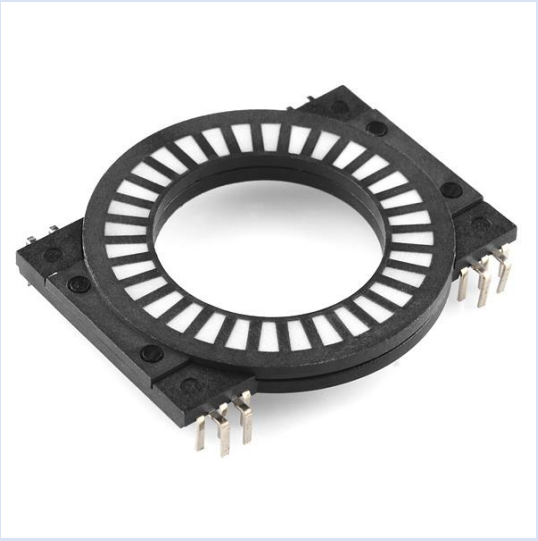
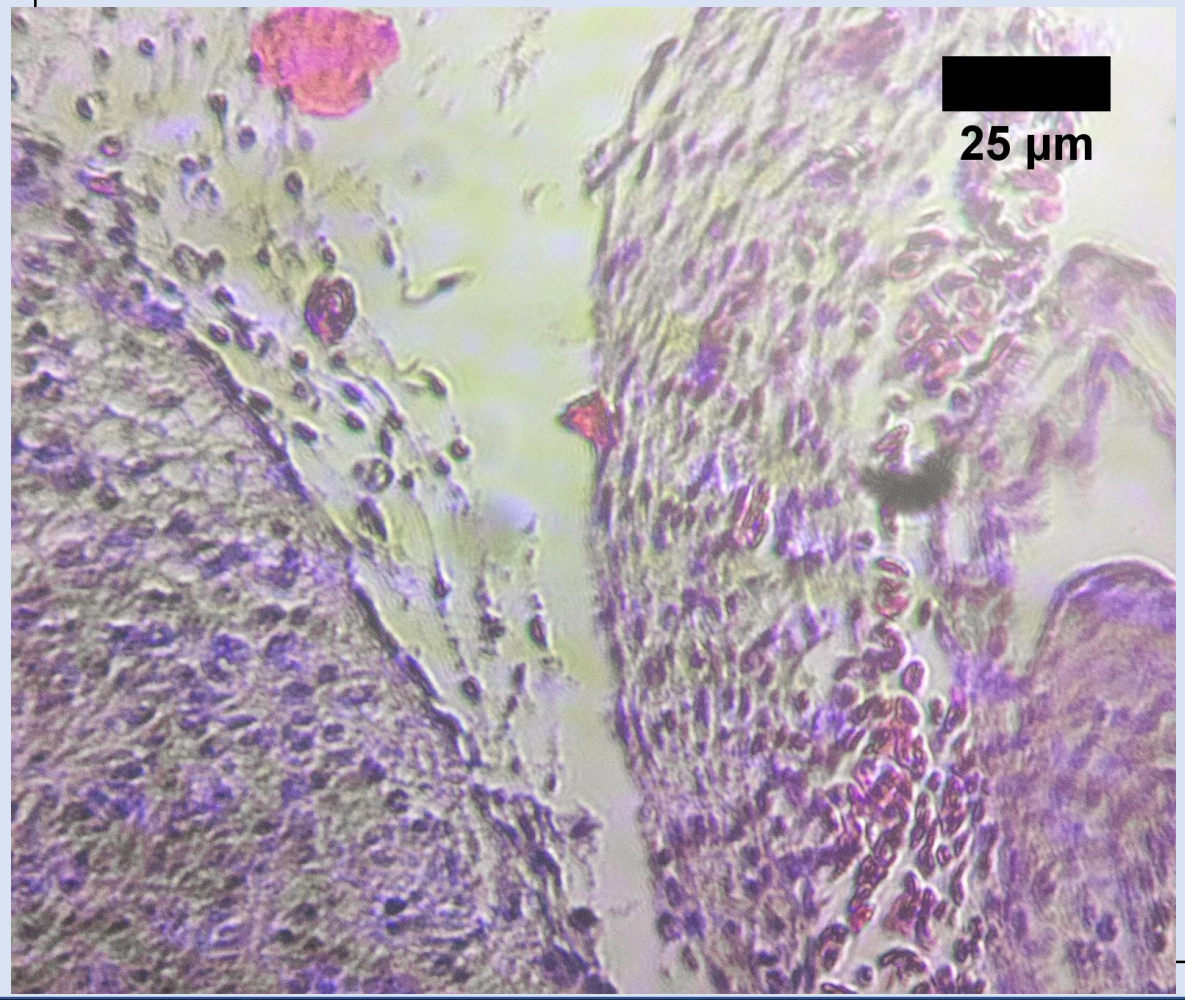
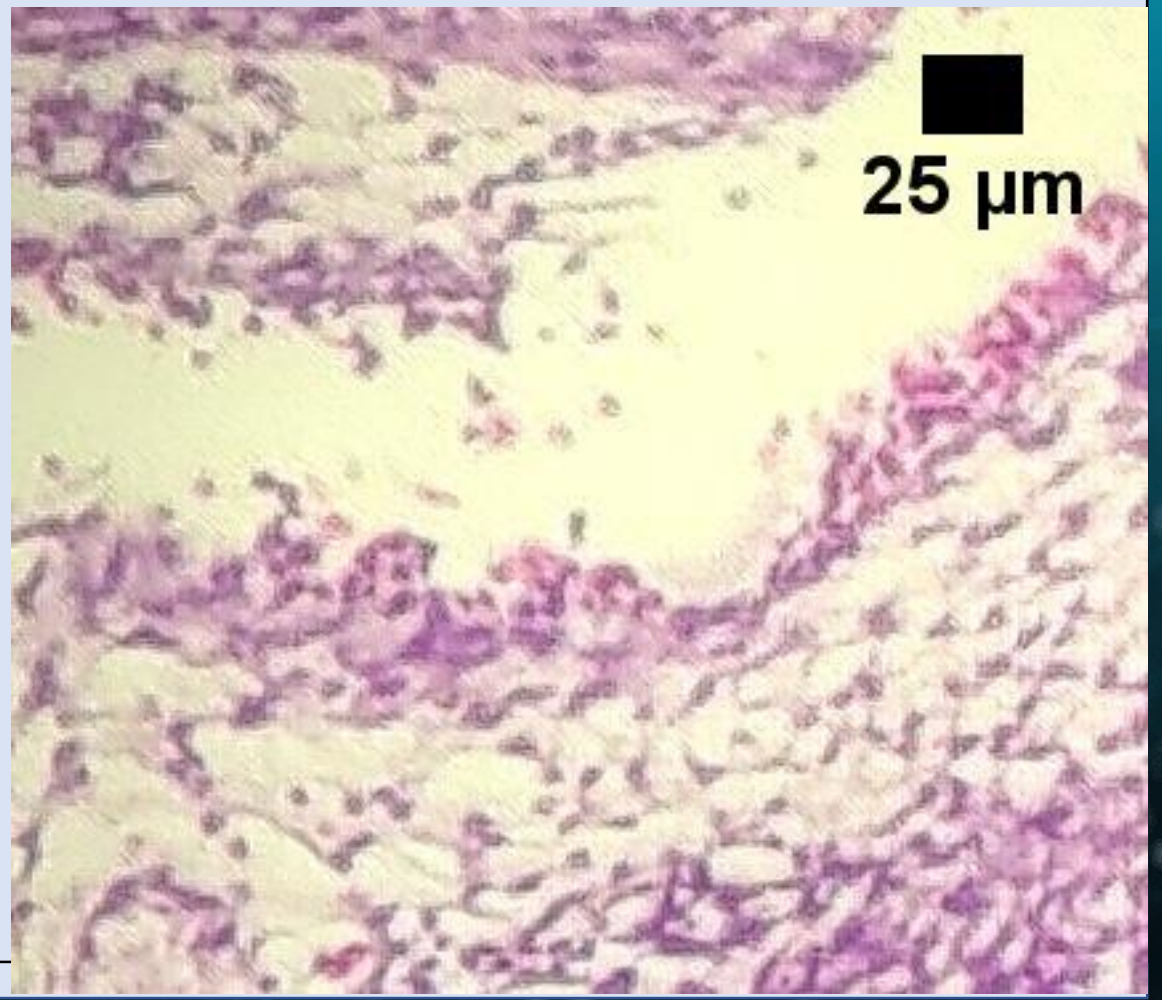


Image Comparison

Design Concept A: 0.11 um/px



Design Concept B: 0.65 um/px



Flutter

Dart

Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Windows, Mac, Linux, Google Fuchsia and the web. The first version of Flutter was known as codename "Sky" and ran on the Android operating system.

Fast Development

Paint your app to life in milliseconds with Stateful Hot Reload. Use a rich set of fully-customizable widgets to build native interfaces in minutes.

Expressive and Flexible UI

Quickly ship features with a focus on native end-user experiences. Layered architecture allows for full customization, which results in incredibly fast rendering and expressive and flexible designs.

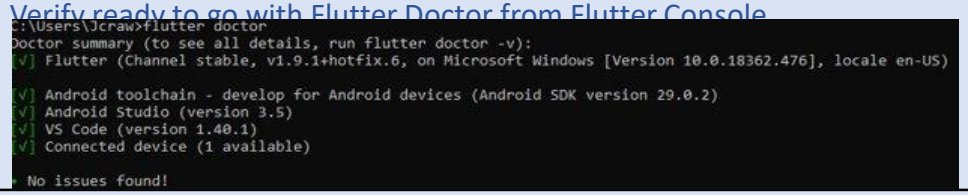
Native Performance

Flutter's widgets incorporate all critical platform differences such as scrolling, navigation, icons and fonts, and your Flutter code is compiled to native ARM machine code using Dart's native compilers. Thus Flutter gives you full native performance on both iOS and Android.

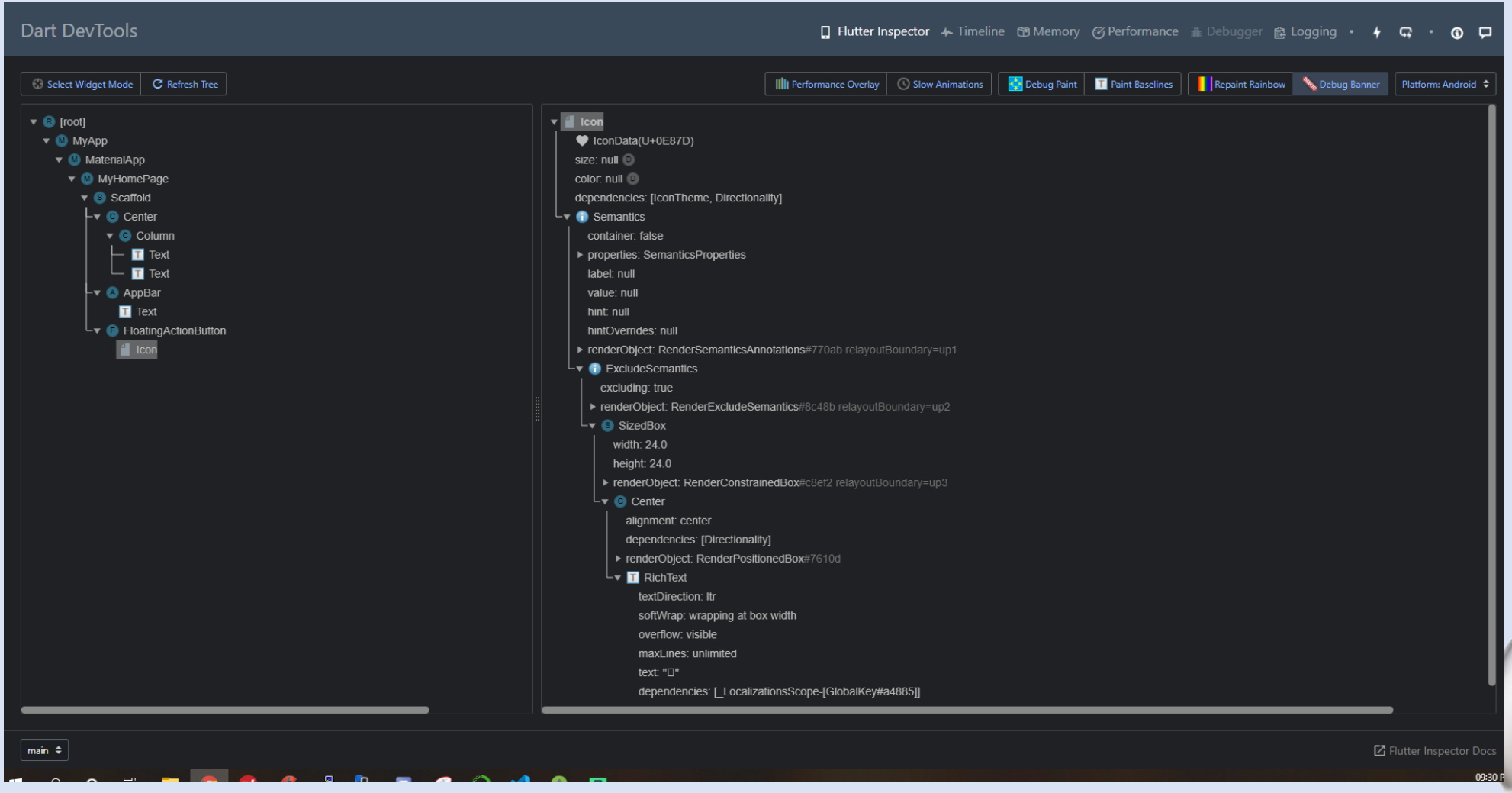
Emulator Setup

Install The following

- VS Code (Ver 1.40.1)
  - Add VS Code to ENV PATH
  - Install Flutter Plugin (3,6,0)
  - Install Dart Plugin (3.6.0)
  - Install Flutter Files (1.5.4) *Scaffold Flutter BLoC Template
  - Install Flutter Help (0.2.5) * VSCode Ext
  - Install Flutter Tree (1.0.0) *Widget Tree
  - Install Flutter Widget Snippets (2.0.0) *Directory of available widgets
- flutter_windows_v1.9.1 env (Flutter Console)
- Add flutter\bin to ENV PATH
- Install Android Toolchain (Android SDK version 29.0.2)
- Android Studio (ver. 3.5)
  - Add Android Studio to ENV PATH
  - Install Flutter Plugin
  - Install Dart Plugin
  - Install Android SDK plugin
- Enable Hypervisor in Windows Features
- Enable Virtual Machine in BIOS
- Verify ready to go with Flutter Doctor from Flutter Console



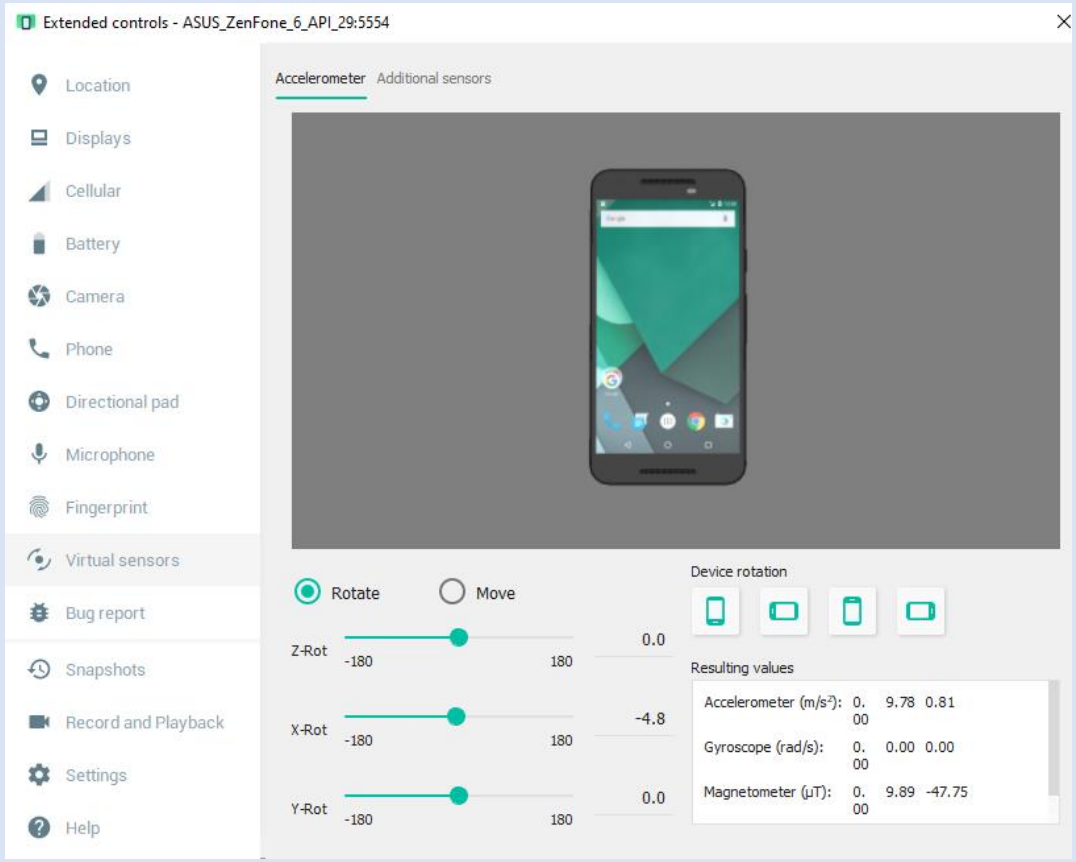
Dev Tools

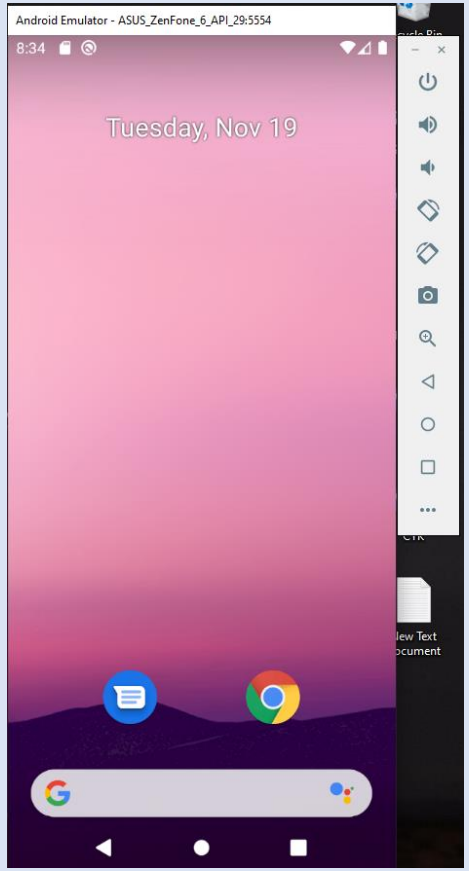

Dart DevTools
Flutter Inspector
Timeline
Memory
Performance
Debugger
Logging
Select Widget Mode
Refresh Tree
Performance Overlay
Slow Animations
Debug Paint
Paint Baselines
Repaint Rainbow
Debug Banner
Platform: Android
[root]
MyApp
MaterialApp
MyHomePage
Scaffold
Center
Column
Text
Text
AppBar
Text
FloatingActionButton
Icon
Icon
IconData(U+0E87D)
size: null
color: null
dependencies: [IconTheme, Directionality]
Semantics
container: false
properties: SemanticsProperties
label: null
value: null
hint: null
hintOverrides: null
renderObject: RenderSemanticsAnnotations#770ab relayoutBoundary=up1
ExcludeSemantics
excluding: true
renderObject: RenderExcludeSemantics#8c48b relayoutBoundary=up2
SizedBox
width: 24.0
height: 24.0
renderObject: RenderConstrainedBox#c8ef2 relayoutBoundary=up3
Center
alignment: center
dependencies: [Directionality]
renderObject: RenderPositionedBox#7610d
RichText
textDirection: ltr
softWrap: wrapping at box width
overflow: visible
maxLines: unlimited
text: "□"
dependencies: [_LocalizationsScope-[GlobalKey#a4885]]
main
Flutter Inspector Docs


APOLLO

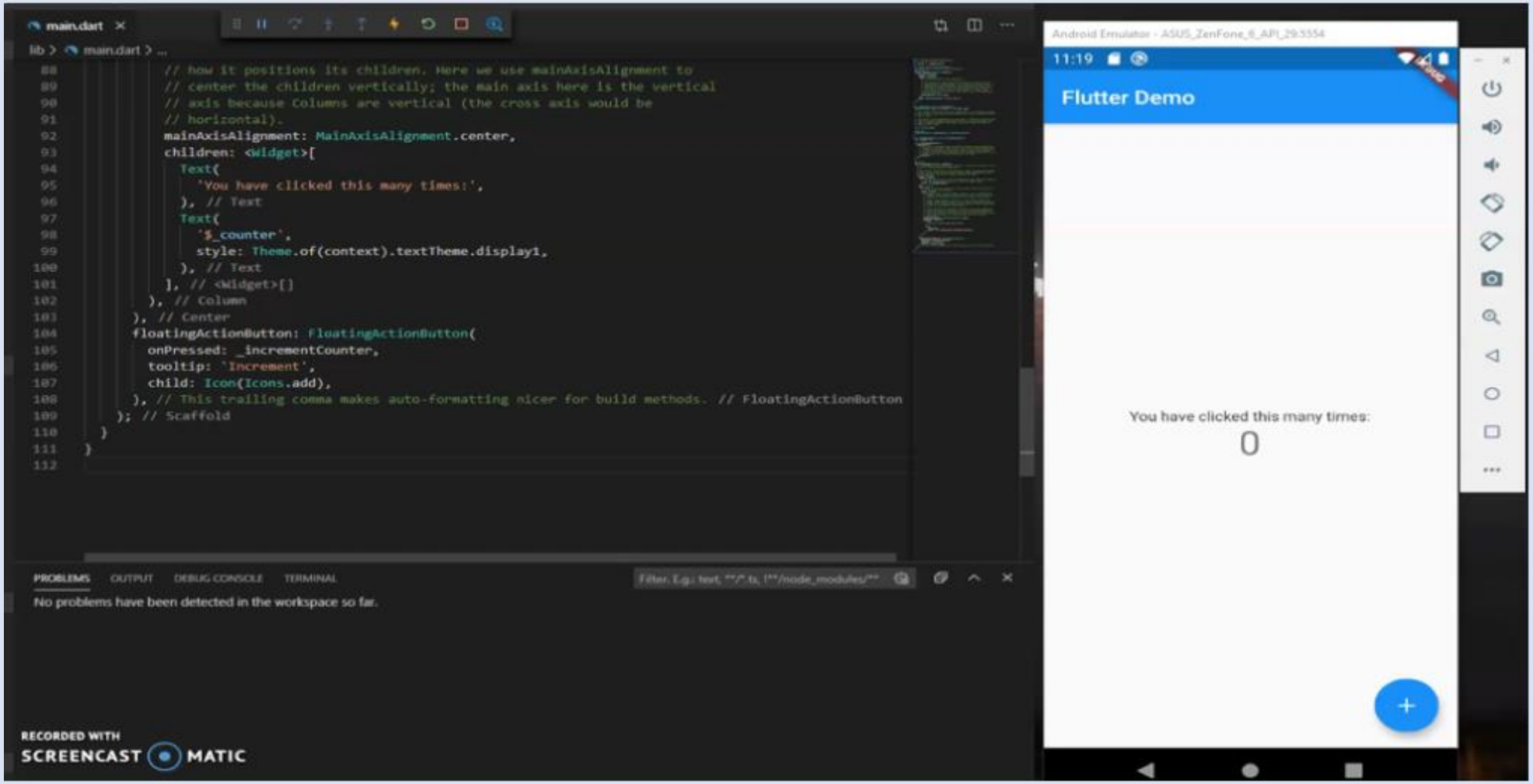
Emulator

- Fired up emulator on Android Studio and create a profile to make.....





Best Part- Near Live Code



Question?

University of Idaho
College of Engineering

Children's Cancer Therapy Development Institute