iOS Easy Keyboard

June 30, 2019

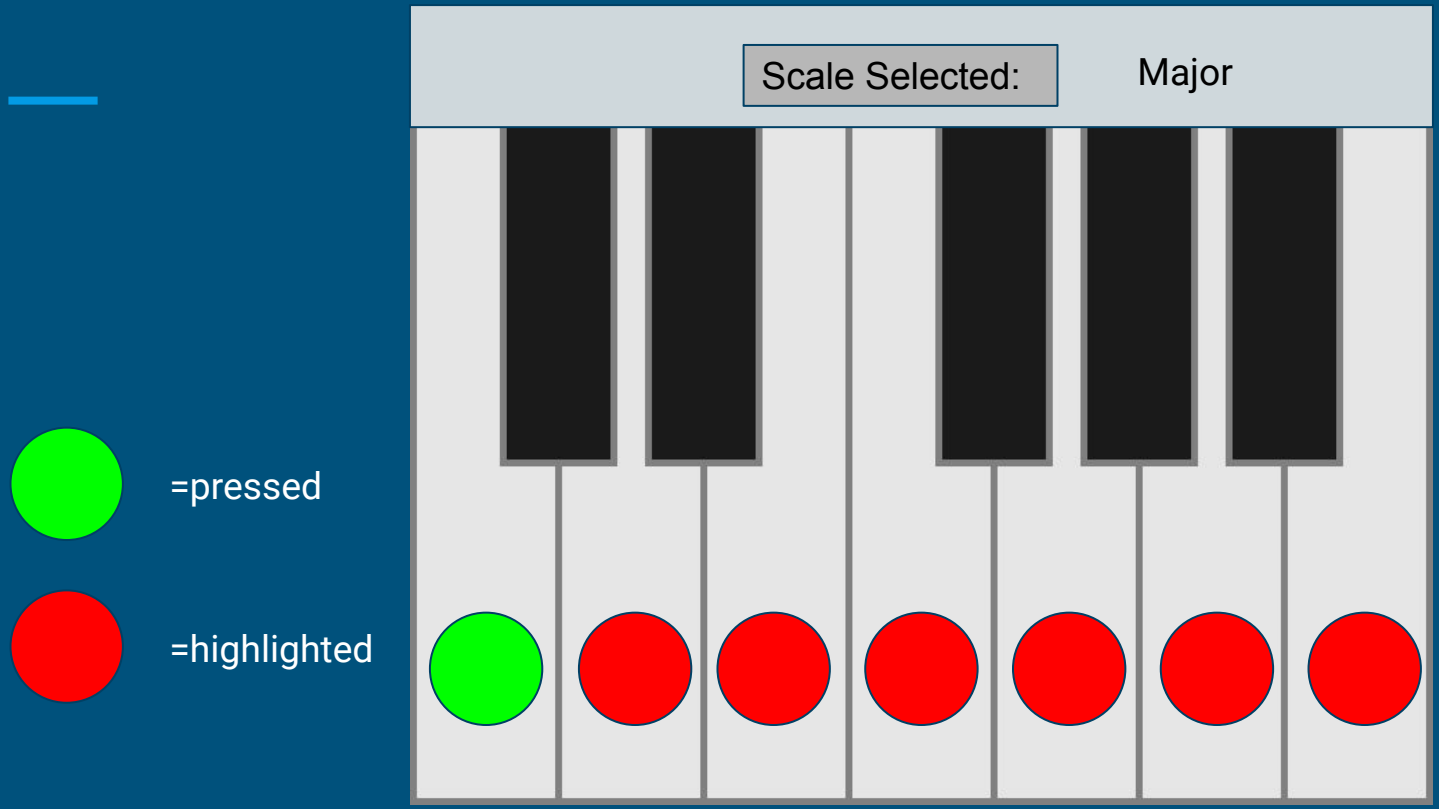
Team: iM Keys
Members:

Arturo Molina - PO
Nithin Rama
Brandon Huss
Sangwoo Kim - 1st Scrum Master

iOS Easy Keyboard

-an iOS app designed to allow users to more easily find the notes in a given scale on a keyboard through visual cues

Concept Image:



High Level Goals

Be able to play a piano on the app (but with limitations of monophonic playing).

Be able to play piano on the app (with polyphonic capabilities).

Be able to display the selected scale or mode on the keyboard.

Be able to select a given mode / scale from a list.

Be able to visually “lock” the scale onto the keyboard.

Sprint #1

User Stories:

1. As a user I want to see a keyboard (or some representation of the 12 tones), so that I can play music. **(8)**
2. As a user, I want the keys to look different when I push them so that I know which key I am hitting. **(8)**
3. Story #2 :- As a user, I want the keys to make sound when I push them so that I can hear what I am playing. **(8)**

Spikes:

-explore audio playback / synthesis for polyphony

Infrastructure:

-setup development environment (Xcode, gitHub, Slack)

Sprint #2

User Stories:

1. As a user, I want to be able to select a scale, so that I can see the notes of that scale highlighted. **(3)**
2. As a user, I want the notes in the key I have selected to be visually distinguished, so that I can find those notes easier on the keyboard. **(5)**

Spikes:

-explore audio playback / synthesis for polyphony

-explore MIDI files and their generation

Infrastructure:

-none

Sprint #3

User Stories:

1. As a user, I want to play chords, so that I can write chord progressions. **(13)**

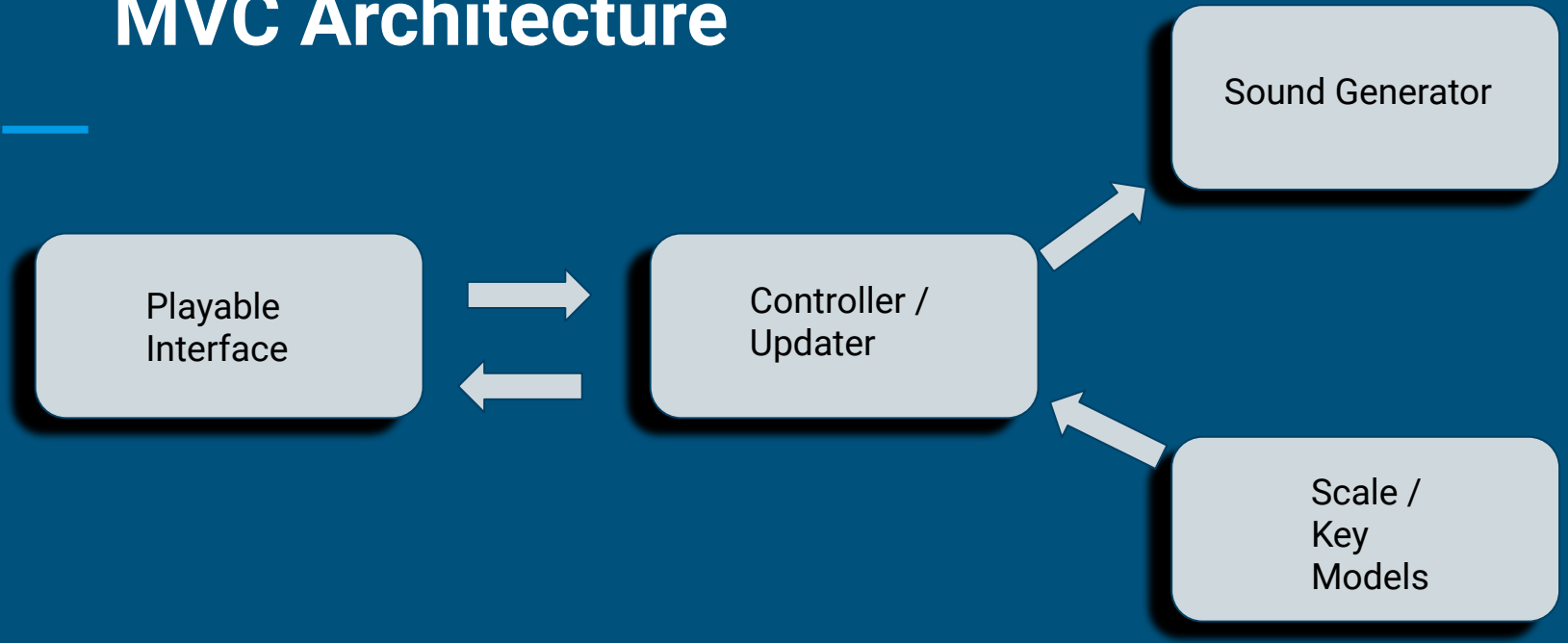
Spikes:

-explore MIDI files and their generation

Infrastructure:

-none

MVC Architecture



Model View Controller Architecture (Models)

-key models that contain information such as pitch frequency

-scale models that contain degrees (intervals) of the scale

-synthesizer or sampler model for audio generation or playback

Model View Controller Architecture (Views)

-view changes depending on user input

-musical interface containing 12 tones

-keys change when pressed

-keys in scale are colored differently

Model View Controller Architecture (Controllers)

-controller gets user input

-retrieves relevant information from the key and scale models

-sends information to sound generator

-sends information from models to view to update keyboard with scales

Challenges / Risks

Challenge #1: Creating custom gui objects that function as piano keys

Challenge #2: Latency. Being able to highlight scales and produce sound in real time, or at least quickly enough to be able to play music with.

Challenge #3: Audio playback

Challenge #4: Polyphony

Technologies

Swift - iOS programming

Xcode - Swift IDE

GitHub - version control / general code management

Minimum Viable Product

A keyboard that plays single notes when the keys are pressed, and is able to display a selected scale visually on the keyboard.