Pocket Choir

Requirements and Specification Document

Kevin Ballecer, Deryk Clary, Shaopeng Feng?, Nicholas Kjorlie, David Kunstmann, Christopher Sullivan

Project Abstract

Pocket Choir is a mobile iOS application designed to provide an intuitive way to live record the user's voice, harmonize it based on a selected chord, and allow for simultaneous editing of effects and parameters. Most polyphonic synths and voice harmonizers available on the app store today do not allow for simultaneous recording and editing features, either focusing more on the synth aspect, or the harmonization feature.

What sets Pocket Choir apart from existent similar applications is its multiple edit/play screens that users can swipe (continuous scroll) or toggle (left/right arrows) between with ease while recording and playing back their voice. Users to can also change the harmonization of their voice in real-time via 8 pads each indicating a different type of chord. The *Edit* screens are customizable, meaning you can choose which effects are displayed and modifiable during live recording sessions. The project's primary focus will be the implementation of a 4-voice polyphonic harmonizer with corresponding customizable edit/play screens, and the proceeding additions of various plugin effects and simple monophonic synthesizer.

Note: The words autotuner, harmonizer and vocoder are each used for a variety of technologies in the music industry. For the purposes of this project, we will be using them interchangeably to refer in general to the audio processes of our product.

Document Revision History: Rev. 1.0 2019-02-18: initial version

2019-03-29: No updates were made to this version of the document.

Customer

We would like to create a system that is intuitive for beginners, but also performs well enough to satisfy experienced musicians. Throughout the planning and design process, potential users will be consulted from both pools. This will allow us to create a balance and gauge how much of a tutorial we may want to build in for new users. Much of our competition could intimidate non-musicians by requiring prior knowledge and piano skills. Being able to easily 'plug and play' will be essential.

Some of our non-musician users were unfamiliar with what a vocoder is. In these cases we described the technology and played example recordings without giving away what the user interfaces look like.

- https://www.youtube.com/watch?v=DnpVAyPjxDA
- https://youtu.be/8XC-8wMuKSs?t=45

Musician User Questions

- What are similar technologies in our area that you use or know of? Would it be realistic to perform with this technology?
- What parts of an interface would you expect and want to use in a vocoder app? (i.e. pitch modulator, keyboard, cutoff knob)
- Would you consider using a mobile device for a performance?
- Would you consider using a mobile device for sampling or music production?

Average User Questions

- What experience do you have using a piano keyboard. Would you say you could play a simple melody given time to figure it out?
- What kinds of music technology or sounds do you find interesting?
- What are the main instruments used in the music you listen to?
- How often do you download and try new apps?
- Do you have an understanding of what musical chords are?
- Which, if any of the following related terms are you familiar with? What other words come to mind?
    - Sample
    - Cutoff

- Looping/overdubbing
- Delay
- Reverb

Competitive Landscape

Currently, there are several mobile iOS applications on the market that act as handheld polyphonic synthesizers made for both professional production and beginner level music-making alike. However, few of these apps achieve an intuitive real-time harmonizer with live-editing capabilities that Pocket Choir promises to provide. The higher end applications serving as virtual synthesizers tend to cost around $10-$30. We aim to focus our product towards those looking for a cheap and easy-to-use harmonizer/synthesizer for beginners and professionals while still retaining high audio output quality. Seeing that a lot of common features amongst existent mobile music synth/harmonizer applications will be included in our product, the focal point of Pocket Choir is its unique interface for live editing and performing. Our product will include scrollable effects/keyboard screens to allow simultaneous editing and playing on a mobile device.

Below is a list of mobile iOS applications that function similarly to Pocket Choir:

Klimper ($8.99)

https://itunes.apple.com/us/app/klimper/id1189322675?mt=8

Features:

- Unique chord-based interface
- Find chords available in desired key/scale
- Find smoother chord progressions
- Basic editing, looping, production tools
- Audio/Midi File Export
- Inter-App Audio Instruments
- 10-built in preset instruments
- Layer chords with multiple tracks/instruments

DRC-Polyphonic Synthesizer (Free)

https://itunes.apple.com/us/app/drc-polyphonic-synthesizer/id973055710?mt=8

Features:

- Polyphonic synth with up to 8 voices
- AUv3 compatible (for use on Garageband, AUM, NS2, Cubasis, etc)
- Two main and two sub oscillators
- Multi-mode filters
- Dual LFO and modeled envelope generators

- Stereo tape delay, reverb, chorus effects
- 4 mode arpeggiator
- Sync external MIDI clock

Harmony Voice ($6.99)

https://itunes.apple.com/app/ivoxel/id532501340?mt=8

Features:

- Four-voice real-time Harmonizer and Pitch Corrector
    - Manual harmonization via chords
    - Auto-harmonization
    - Chords auto added to any base note
- Select key and scale for pitch correction and harmonization
- Edit voice
- Play background tracks from iTunes
- Reverb, Chorus, Delay effects
- CoreMIDI compatible
- Audio Recorder w/ Metronome
- File export and direct upload to Soundcloud
- Audio clipboard for inter-app exporting

User Requirements

1. **Home Screen Loads**
    a. *Create New Project*
        i. Chords Screen (refer to 2)
    b. *Load Existing*
        i. Select a file previously created in-app
            1. Chords Screen (refer to 2)
        ii. Load a sample from iTunes library or audio sample in File System
            1. Chords Screen (refer to 2)
    c. *View Instructions*
        i. Browse through various text and video tutorials using different freatures in-app.
2. **Play/Record/Effect Scrollable Screens**
    a. *Chords Screen*
        i. 12 keys to play different types of chords
        ii. Octave Lower (left arrow)
        iii. Octave Higher (right arrow)
        iv. Swipe Left = Customizable Effect Page 1
        v. Swipe Right = Recording Screen

b. *Compressor/Chorus/Overdrive (Customizable Effect Page 1)*
    i. Change Compression
        1. *Attack* slider
        2. *Decay* slider
        3. *Sustain* slider
        4. *Release* slider
    ii. Change Chorus (*Chorus* knob)
    iii. Change Overdrive (*Overdrive knob)*
    iv. Swipe Left = Customizable Effect Page 2
    v. Swipe Right = Chords Screen
c. *Synth Modulation/Speed/Filter LFO (Customizable Effect Page 2)*
    i. Synth Mod
        1. *5* wave sliders - change wave of oscillation
        2. *Noise* slider - control amount of noise heard in sound wave
    ii. *Filter LFO* (Wah) knob
    iii. *Speed* knob - adjust speed/pitch of the sample being recorded
        1. Not available during live recording
    iv. Swipe Left = Global Parameters Page
    v. Swipe Right = Customizable Effect Page 1
d. *Global Parameters Page*
    i. *Master Volume* knob
    ii. *Cutoff Frequency* knob
    iii. *Resonance* knob
    iv. Swipe Left = Recording Screen
    v. Swipe Right = Customizable Effect Page 2
e. *Recording Screen*
    i. Tap *Record Sample* to start recording your voice
        1. Tap *Stop Sample* to stop recording of voice, sample is now automatically mapped to chords on chord screen
    ii. Tap *Begin Recording Live Audio*
        1. Users can now simultaneously record and tweak harmonization and other effects to the desired levels
    iii. Sample Editor
        1. Hold down on the sample for 3 seconds to make end bars appear
            a. Move the end bars to select the desired sample length
            b. Double tap to make the bars disappear
        2. Drag the sample to select desired sample section
f. *File Export* (Tap Upload button)
    i. Export file to:

1. Dropbox, Google Drive, Mail, Messages, Files, Garageband, etc.

g. *File Save* (Tap floppydisk icon)
    i. *Name File*
h. *Record* (Tap red button) - opens the recording screen without having to scroll
i. *Load* (Tap + button) - opens the *Load Existing* page

3. **User settings**
    a. Ability to select input/output audio source
    b. Ability to select default export location
    c. Ability to select different plugins/effects to for customizable effect pages

Interviews of Potential Users

Each member of the team has reached out to at least two potential users of this product to gain insight on what type of interface and what features would make our autotuner stand out from the competition. All expressed interest in testing the product once a functional demo is created. Our motivations are to create a harmonizer that is both easy to use for beginners and powerful enough for an experienced musician to perform with. While most of our interviewees were casual users, we found three musicians with experience using professional vocoders and synthesizers to help us understand key industry standard features to include. Many of the questions we asked are listed above in the customer section. The results of our user interviews have been compiled and consolidated into the following user stories.

User Stories:

- When I press a piano key and sing into the microphone, my voice is tuned to that note.
- When I click a chord and sing into the microphone, my voice is simultaneously tuned to the different notes in that chord
- I can choose to output the audio through internal speakers, external speakers, or headphones
- I can choose to use an internal or external microphone
- I can modify the autotuner's sound with cutoff, vibrato, and other effects while playing
- I can alter the interface so my commonly used effects are quicker to access
- I can record a sample in the vocoder and export it as an audio file
- I can save a settings configuration for later use

Use Cases

Homescreen

Name	Opening a sample file
Actors	User
Triggers	“Open sample file” is selected by user
Events	● a list of previously saved files are displayed ● if a file is selected, go to play/scroll screen
Preconditions	User is at home screen / there are saved files
Acceptance Test	Save a file and reopen the application. Attempt to open the previously saved file.

Name	Creating a new sample file
Actors	User
Triggers	“Create new sample file” is selected by user
Events	A new file is created and the play/scroll screen is displayed
Preconditions	User is at home screen
Acceptance Test	Check that new file is on all default settings.

Play/Scroll Screens

Name	Recording a sample
Actors	User
Triggers	User taps on record icon
Events	A sample is recorded and harmonized/autotuned to selected chord in real time
Preconditions	User selects chord to harmonize with
Acceptance Test	Try navigating to different scroll screens and see if recording continues. Try different effects.

Name	Saving a sample
Actors	User
Triggers	Save button is selected from top bar

Events	If previously saved, can be overwritten or if not, name and save new file
Preconditions	A project is open
Acceptance Test	Reopen application and attempt to load previously saved file. Check that all presets are loaded.

Name	Exporting a sample
Actors	User
Triggers	Export button is selected from top bar
Events	User selects which platform to export to
Preconditions	A project is open
Acceptance Test	● Check that file is properly formatted for destination ● Check audio quality

Name	Changing bass note/chord
Actors	User
Triggers	A bass note/chord is selected from keyboard
Events	Current sample will be harmonized with bass note/chord
Preconditions	User is on keyboard screen
Acceptance Test	● Perform a simple sound test/comparison by ear ● Test with all bass notes/chords

Name	Switching between audio effects
Actors	User
Triggers	Screen is swiped left or right
Events	If another effect is available, that effect is locked onto the screen and the parameters for it are displayed
Preconditions	User is on home screen or on an effect page
Acceptance Test	Turn on all effects and scroll through each, testing all possible

	configurations/parameterizations in a live recording session

Name	Changing parameters of an audio effect
Actors	User
Triggers	If a knob for an audio effects parameter is moved
Events	Parameters for relevant effect are manipulated in real time
Preconditions	User is recording or a prerecorded sample is selected
Acceptance Test	Try all possible parameterization configurations on all effects

Name	Edit a sample
Actors	User
Triggers	● Cropping sample by dragging the edges of the waveform ● Speeding or slowing sample by selecting one of several speeds
Events	A sample is either cropped, sped up, or slowed down
Preconditions	User is on sample editing page
Acceptance Test	Crop or change speed of sample. Save and reload the sample to see if changes are still in effect.

Settings and Help Menu

Name	Settings menu
Actors	User
Triggers	User selects settings icon
Events	● Audio effects can be chosen for display on home screen ● Audio input/output configuration
Preconditions	Settings icon should be accessible from anywhere in application
Postconditions	Settings are applied and saved upon exiting settings menu
Acceptance Test	● Try accessing settings menu from all different interfaces ● Check audio input and output with desired device configuration

Name	Help menu
Actors	User
Triggers	Help icon is selected
Events	A read-me as well as video tutorials about how to interact with the user interface is displayed
Preconditions	User is on homescreen
Acceptance Test	See if text is readable and that videos are playable

System Use Cases

Name	Harmonizing
Actors	Backend of the application
Triggers	A bass note or chord for harmonizing is selected while recording a sample
Events	Backend converts audio input from microphone into a harmony
Preconditions	Sample is being recorded or a sample is loaded
Acceptance Test	See if converted audio matches what was selected on the front-end interface

Name	Applying audio effects
Actors	Backend of the application
Triggers	An effect is enabled from the homescreen or an effect parameter is manipulated
Events	Backend converts audio in accordance to the change of parameters and is output in realtime
Preconditions	A sample is being recorded or has been recorded and is selected for editing
Acceptance Test	See if converted audio matches what was selected on the front-end interface

Name	Saving a project

Actors	Back and front end of the application
Triggers	User selects save option
Events	Backend stores formatted file with a user specified name
Preconditions	A sample is being recorded or has been recorded and is selected for editing
Acceptance Test	• Check if name and file type that is stored in the backend matches what was selected on front-end • Attempt to load file

Name	Loading a project
Actors	Front and back end of the application
Triggers	User selects to load previously saved file
Events	Front end retrieves user specified file from back end
Preconditions	A file has been previously saved
Acceptance Test	File returned from back end matches what was requested from front end

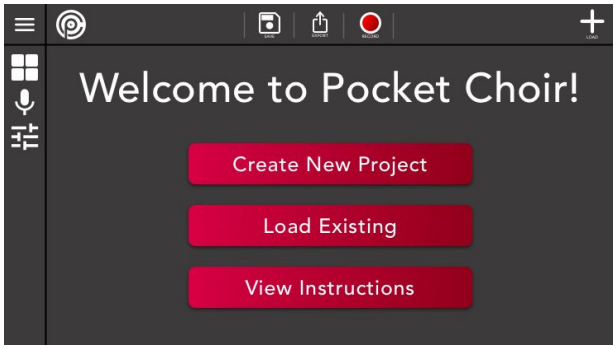
Name	Exporting a project
Actors	Backend of the application
Triggers	User selects export option
Events	Backend converts file to format specified for source of export
Preconditions	A project is currently loaded
Acceptance Test	Open file at the specified destination and check audio quality and integrity

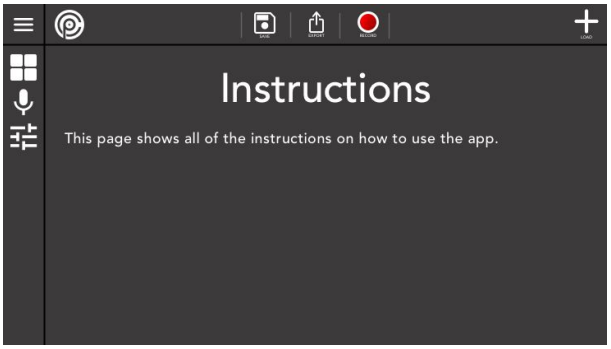
User Interface Requirements

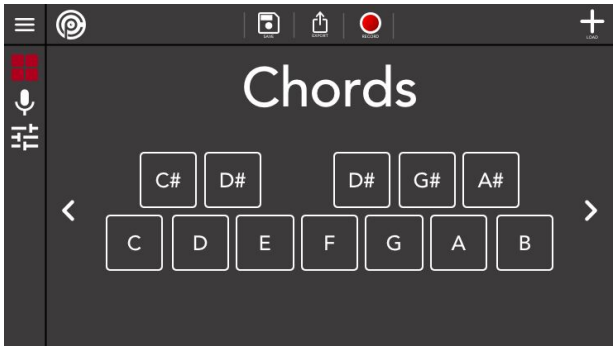
The user interface of our app has to accommodate the small screen of a mobile device while still allowing for maximum functionality. This mean all tools must be large enough for the users to click what they want without hitting undesired buttons and the app itself needs to be organized well enough that they can get to what they want quickly, especially when using the app with live audio. To overcome these obstacles we have two toolbars: one along the top of the screen and one along the left side, as seen in the pictures of the user interface mockup below. The toolbars contain shortcuts to the tools the users need in a timely manner. These shortcuts include the chord page, the recording page, the tools page with the oscillator, compressor, and other effects, and the save and export menus. We even included a button on the top toolbar to instantly begin inputting sound from the device's microphone and harmonizing it.

With that, the only input needed from the users is the audio they would like to be harmonized and distorted with the tools provided by the app as well as their desired settings (harmonized chord and other effects). The output the users need is the harmonized audio either fed through the speakers or headphone jack, or saved to a file on their phone/in the cloud (DropBox, GarageBand, iCloud, Google Drive, etc.). When saving to a file, we will provide users with the option of compressing to a lossy format like mp3 or exporting to FLAC (Free Lossless Audio Codec). This decision is based on what users deem most important: file size or audio quality.

User Interface Examples







Security Requirements

User Data

As we will not be asking our users to create accounts, it won't be necessary to encrypt or otherwise secure any emails, passwords, or other personal information from the users. We will however need to secure any audio samples or recordings the user creates in a local database that is not accessible by other apps and services.

Phone Hardware

Our application will make use of the touch screen input, microphone input, and the audio output hardware on the user's device. The primary vulnerability to consider will be the microphone. An unforseen backdoor into our vocoder processes could grant a malicious party access to not only audio generated in the app, but potentially the private conversations of the user. Securing the microphone access will be critical in developing user trust in our system.

System Requirements

SDKs and Libraries

- AudioKit (Swift audio processing library)
- Built in iOS GUI
- AudioKit GUI

Data Protocols

- MIDI

Data Capacity and Memory Requirements

Memory and processing minimums are currently unknown. We suspect that a certain threshold will need to be met. Without enough resources, the user's vowels and pronunciation will not be accurately translated to different pitches. Intelligible lyrics are critical for most vocal processing technologies.

Device Requirements

- iPhone/iPad/iPod Touch running iOS 11.0.0 or higher
- No data, wifi, or bluetooth is required to run this app once downloaded to a device
- External speakers or headphones are necessary to avoid feedback with the microphone

Specification Activity Flow