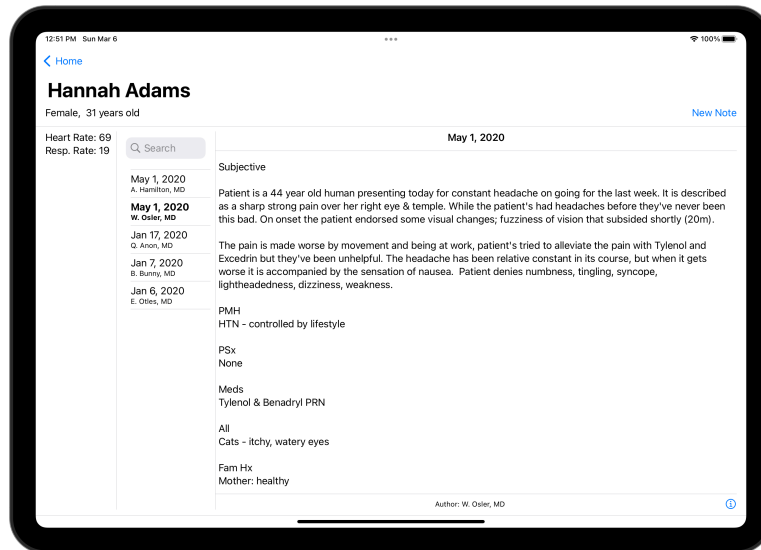# Doctor's Notes Software Prototype

Erkin Ötleş
March 7th 2022

Topics: health information technology, doctor's notes, electronic health records, software design, user interfaces, user experience, human computer interaction

The iPadOS app I made to test out new functionality for doctors writing notes.

We will return to the "Intro to ML for Physicians" series next week. In the intervening time here's a short post about a prototype health IT app I made a two years ago. I made this app as part of a team project that was focused on examining and improving the way doctor's notes are written.

Nominally this was a graduate project (holler at my HCI team[1]) and the project specification called for making a low-functionality prototype using *invision*. [1], We did this and found it unsatisfying. The reason for this was that we wanted to incorporate a voice transcription interface into the note writing process. Although we could replicate some of the other functionality there was no way to build voice transcription and other key functionality in the prototyping software.

So I took the logical nextstep[2] and built out a minimal viable prototype using Apple's development tools. This allowed me to incorporate on-device transcription. [2, 3] On-device transcription is a really cool technology for healthcare IT! Because you don't have information

---

[1]Sarah Jabbour, Meera Krishnamoorthy, Barbara Korycki, and Harry Rubin-Falcone

[2] kudos if you got the joke

flowing off the device back to Apple's (or someone else's) servers, it could enable HIPAA compliant voice interfaces in the future. Making a prototype app also enabled me to build several other features, such as saving and retrieving notes. These features are necessary when testing out a more complicated record keeping system, like this.

If you are interested in learning more about this prototype check out this video:
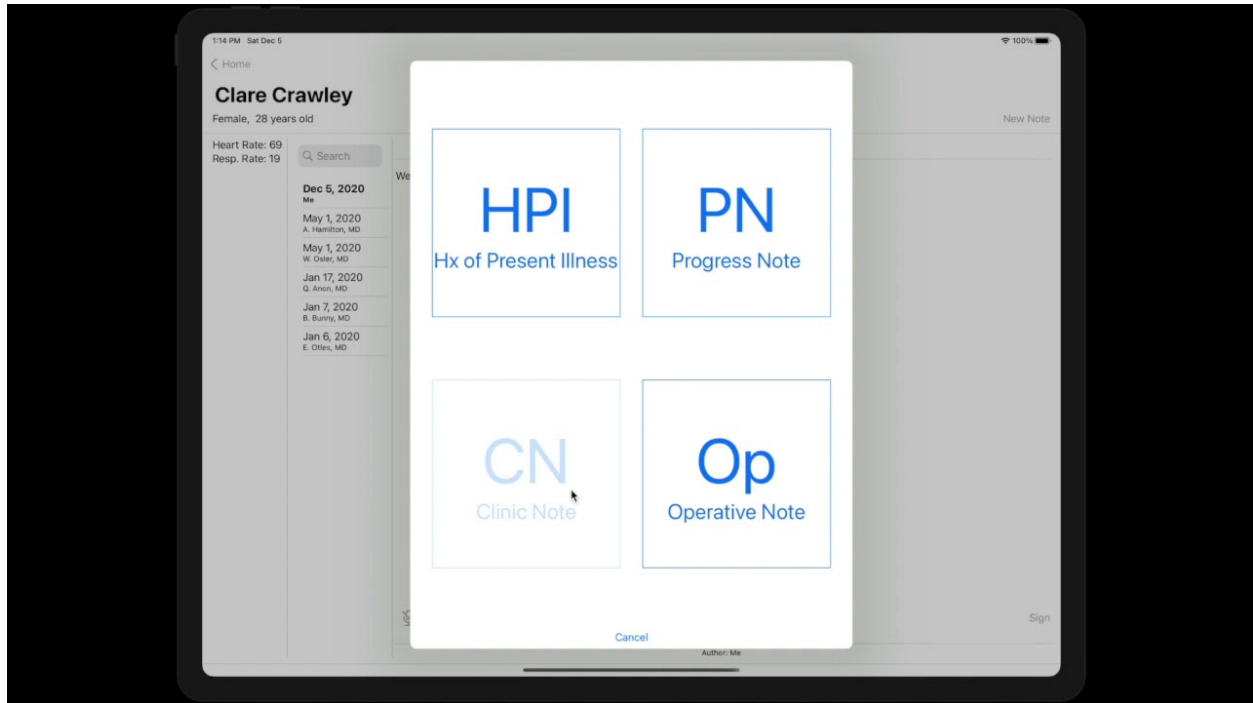
If you would like to take a look at my hacky Swift code check out the Github project.

One thing that I didn't have time to code up was the sharing of notes between physicians. This is a pain point in systems that are actually in use. The team had some cool ideas about collaborative editing and version control. I think these would be super useful from both a clinical perspective (making the sharing, editing, and co-signing easier) and also from a technical perspective. However that would involve a significant amount of back-end development (see: Complicated Way You See Patient Data: EHR Front-Ends so it remains an item todo.

One of my mantras is that there's a lot of work to be done in healthcare IT. Developing prototypes and testing them out can help us advance the state of the field. Rapidly prototyping these systems is hard to do, but it could pay dividends in terms of physician happiness and productivity.

Erkin

P.S.

Although I've made a couple other apps using Xcode and Swift this was my first time using SwiftUI, which was a pretty slick experience.[4] I really enjoyed programmatically creating the interface and not having to toggle back and forth between my view controller code and the Interface Builder.

Acknowledgements

I'd like to thank the team: Sarah Jabbour, Meera Krishnamoorthy, Barbara Korycki, and Harry Rubin-Falcone. Making wireframes with you guys was an absolute joy.

Bibliography

1.   *Prototype | InVision*. 2022; Available from: https://www.invisionapp.com/defined/prototype.
2.   Bolella, D. *SpeechTranslatorSwiftUI Github Project*. Available from: https://github.com/dbolella/SpeechTranslatorSwiftUI.
3.   *Recognizing Speech in Live Audio | Apple Developer Documentation*. 2022; Available from: https://developer.apple.com/documentation/speech/recognizing_speech_in_live_audio.
4.   *SwiftUI Tutorials | Apple Developer Documentation*. 2022; Available from: https://developer.apple.com/tutorials/swiftui.