

Team Scrubs - Part 2

(Abhi Malhotra, Zach Richmond, Pranil Vora, Brad Ware)

I. Description

As some of you may know, it can be a nightmare traveling to a different country. Not only do you have to worry about the trip, but health regulations and your safety while traveling can be a major concern. Not to mention that some requirements are constantly changing due to current events in the region one is trying to visit.

Team Scrubs is designing a mobile application that allows people traveling abroad to access their health information and also the requirements of the destination. Our application will give users the most up to date health and safety information for every one of their destinations, while also giving them easy to understand instructions and tips for everything from preparing to travel to precautions they should take after coming back. In the unfortunate case that someone does fall sick abroad, their entire medical profile and history will be accessible through our application for quick reference. Having personal health data at your fingertips while abroad can prove to be lifesaving in the event you need emergency medical attention.

Because our application is used for utility and quick information on the go, we thought it would be best to stress it's simplicity so that it can be navigated while 'running through an airport'. Every function needs to serve a practical purpose and a solution for current medical problems faced by those who travel. We emphasized simplicity through the amount of pages the mobile app needs, because there were just a few core actions. The structure of the application's layout is based on the 4 core functions of the application.

The first core function is the *Login*, where the user will put in their credentials and gain full access to the application. Security is of the essence because we are storing very critical private information. The second function is *Profile*, where as you may have guessed the user manages some of their broad contact information. This may include name, phone, address, doctor, and emergency contacts. The third function is *Trips*, where the user can manage their trips abroad. Some of the associations on this page would be the country, destination airport, dates, and current health news. Lastly, the fourth function is *Records*, where one keeps detailed information about their medical history. This can include vaccination history, previous and current doctor, prescribed medications, and proofs of information. This page is critical to manage your personal medical history, and also cross reference it to what is necessary by the requirements of the destination country.

II. Design 1

For design 1, the team decided to use a traditional way to organize information on a mobile device. Navigation bars are very common in interface layout, as they provide a very practical way to display options to the user. On mobile this becomes a little more difficult due to the smaller sized screens.



top left: Login, top middle: Profile, top right: Records, bottom: Trips

However, with design 1 we decided to use a traditional style of showing a hidden nav menu by representing it with the three horizontal dashes stacked on top of each other. This is used by many mobile applications to alert the user that a nav menu exists if you expand the hideaway icon. This hideaway menu allows the option for each of the 4 core functions that our application permits for the user.

Strengths

Let's first walk through this particular design and the strengths associated with it. This design is organized by the hideaway vertical menu, which gives the user access to any page no matter where they currently are in the application. Thus, the user will never feel trapped or inside a hierarchy of actions where they have to undo previous items to get to the original page. The affordance for the menu is a recognizable symbol that many mobile applications use. This will not only make it familiar to the user right away, but will provide a conceptual model for most of the users of this application. Overall, this design works great for users on the go or high stress situations, because they can easily reach every core function of the application from any page. It also saves real estate space on the smaller mobile size screen because it is a hideaway menu. Therefore, it still retains effectiveness while allowing the most information on screen at once.

The *Login* page is pretty standard of typical applications. One major strength is that not only does the user need a name and password, but also a pin ID in case they step away from their phone. Because this application contains such private information to the user, it is essential that an extra security measure is included. It also includes the ability to perform an emergency call even if you are locked out of the application. If someone finds you unconscious or in danger then they can choose that call which will alert your emergency contact, and also the closest hospital near your travel location. Both of these can be found within the application itself.

The *Profile* page has the personal information regarding each user, and allows editing inline. The strengths of this design is that you can quickly update your information within the application. The collapsible dropdown style also allows the page to display every category clearly to the user, and then more information can be expanded if needed. The plus/minus symbols are great signifiers for the state of the dividers, and provide great affordances to the user to inform them that more information is available under each category.

The *Records* page has a very similar design to the *Profile* page. The strengths are the affordances of every category under the *Records* section. It also provides the signifier for the collapsible option which is a standard among web and mobile interfaces.

Finally, the *Trips* page is again similar to the *Profile* and *Records* in terms of layout. Every major trip will have its own section of the interface with a collapsible dropdown menu of minor details. A strength of this design is that the information is organized by which trip you took, so that information is not confusing or misinterpreted. It clearly lists the date of the trip, the destination airport, and all of the medical requirements and current events of that region. Therefore, the user has to only focus on one region at a time, where this provides a conceptual mapping to how most people plan and think about a trip. Each instance is independent of the rest in the interface.

Overall, one of the major strengths of this design is that each core function has a very uniform interface. The user will know how to interact with each of the pages because they all follow the same design guidelines. This design also uses conceptual mapping for menus and navigation that is extremely similar to other mobile applications.

Weaknesses

This design does contain some weaknesses. One of the major overall flaws is that information is displayed inline, which can cause styling complications for inputting information. It also does not take advantage of the iOS's horizontal nav bar at the bottom of the screen. This is a standard among iPhone applications and is what users look for when trying to decide about possible actions to take. However, let's explore some of the weaknesses for all the core functions.

The *Login* page follows a pretty standard format of most interfaces. However, one design flaw is that users could be confused by the emergency call button. How would they know who it will call? What if they are not familiar with the application? How can we override the device's operating system to display that functionality throughout the phone, instead of just on the application's home screen? In a true emergency, I doubt anyone would take time to unlock the phone (if they knew the password), navigate to the *Srubs* app, and then press emergency call button. It takes too long and is unfamiliar to users, especially those who know nothing about the *Scrubs* app.

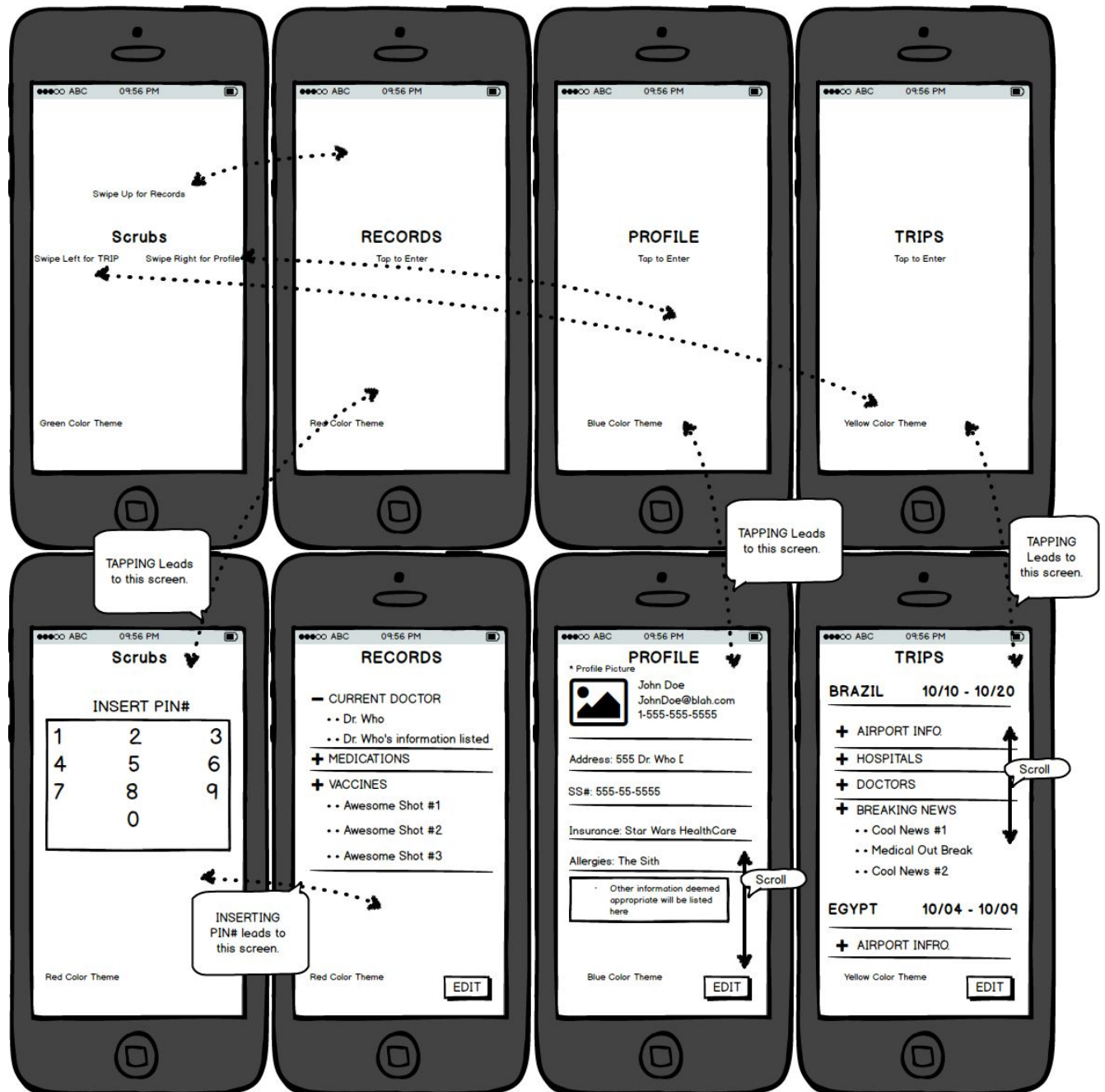
Because the *Profile*, *Trips*, and *Records* pages contain such similar interfaces, we want to analyze all three of them together. We think the information is grouped very naturally, and makes it easier to manage a user's information. However, inline editing can be very tedious on a mobile phone. Because the screen size is so small, it can be difficult to see the information inputted currently, and the current changes the user is making. Most application's get around this by having a desktop version for inputting new information, which may be something *Scrubs* has to explore. Because each user is inputting private and very sensitive information, they need to be sure about the validity of the information that they are adding or editing. An alternative would be to link each category to its own *edit* page, where just the form is shown on the screen. This would maximize screen

space to allow users to input information of all size ranges. Another flaw might be the location of the *edit* in the bottom right of the screen. Because users scan the screen just like the pages of the book, they first start at the top left and cycle through the page ending at the bottom right. This might make it frustrating for users on the go that quickly need to edit their information but can't find the *edit* button to activate that mode. An alternative would be to place it in the top left of the content pane, to ensure users see it first before scanning the actual content.

Even though this interface is simple and provides a conceptual mapping of other applications to key actions, it still has some confusing aspects that need to be addressed. How do we alert the user what the *emergency call* button does? How do we make it accessible if you are locked out of the phone? Should the edit button be placed in the bottom right corner? Is inline editing realistic for the extensive personal content that our applications contains? Each of these questions need to be addressed in the final design of the *Scrubs* application.

III. Design 2

The second design we had for our application includes swiping to get to different pages of the application. We were inspired by this design because it gives the application a sleek minimalist look and provides a neat way to navigate to every core function of *Scrubs*. The mockups for the design can be found on the next page, as well as the breakdown for the strengths and weaknesses of this design.



Strengths

The main strength of the swipe design is that it takes advantage of a well known conceptual model that *Snapchat* uses and then builds off it. When you open the app, you're taken to the homepage *Scrubs*, then the user can swipe left, up, or right follow by an accessing tap to continue on to Trips, Records, or Profile. This model assumes you are already logged in, it additionally adds a pin # to access important medical files/information, which is another strength of the app. It is important to note that this information on the "Records" page is personally uploaded, it does not sync up with a Doctor's office, it does contain a Proof section for the user to upload validation documents backing the user's inputted medical records.

This design takes advantage of all four Gestalt Principles and contains redundant color coding. The app uses closure on the "Inserting PIN#" page and continues to use it throughout the app by adding horizontal lines to separate pieces of information.

On the *Records* and *Trips* pages, the design uses common fate and similarity together to display display information in a typographical hierarchy, which is usercentered. Additionally, these pages take advantage of a well known conceptual model of *plus/minus* (tap the plus sign to show all the information and tap the minus sign to hide all the information) to display information. Proximity is used in the sub bullets listing information, it is obvious the information is connected and belongs together.

The *Profile* page uses closure to separate personal information about the use while identifying each piece of information by labeling what it is, which takes advantage of proximity.

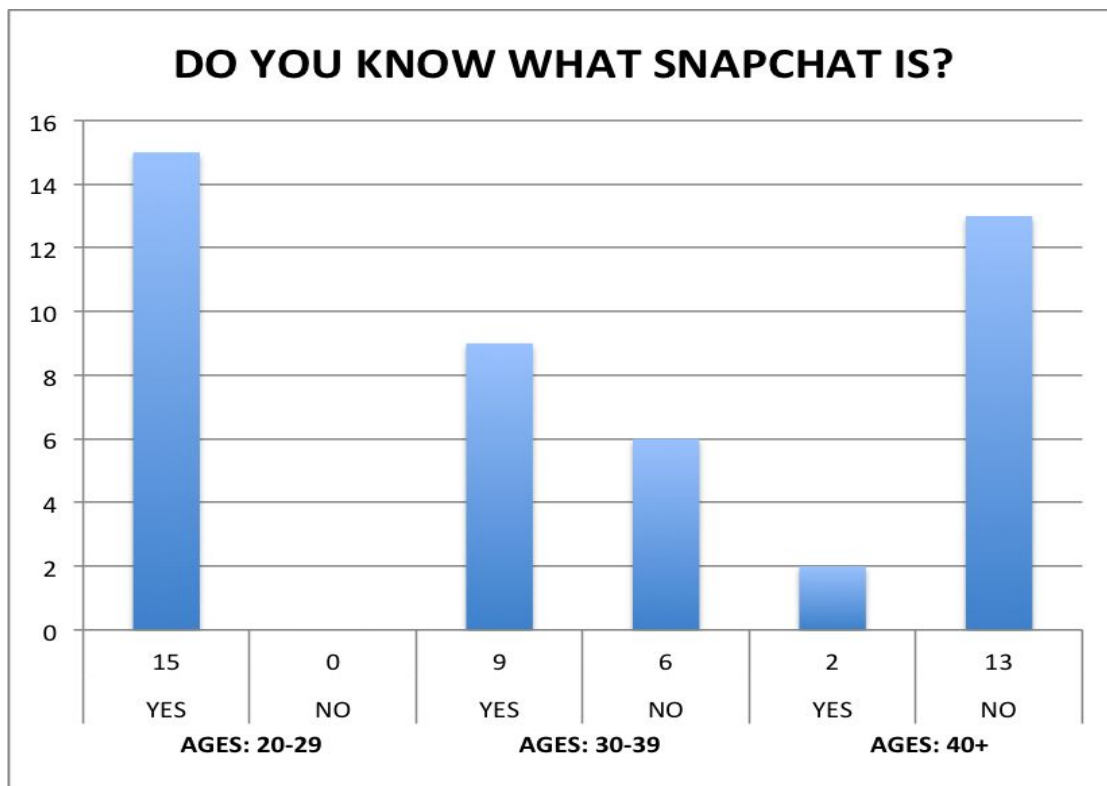
The final strength of this design is the redundant color coding, in the bottom left hand corner of each page design there is color corresponding to the page. For instance, anything to do with *Records* will be red.

Weaknesses

The main weakness of this design is also it's greatest strength, which is taking advantage of the Snapchat conceptual model. Shown below in the graph attached, one can easily see a trend directly to how old the user is and their knowledge of Snapchat. This might create an issue for the older group of the target audience because accessing any of the pages involves swiping, and this functionality is not 100% visible. A solution for this issue would be to notify first time users on how to use the app or add a help button at the bottom of the home screen.

Another weakness of this application is how information will be dispersed on the *Records* and *Trips* pages. Although it takes advantage of the *plus/minus* feature for consolidating information, this might become cumbersome if a header has many pieces of information. The user would be scrolling up and down for a while to arrive at a single piece of information or to minimize a section. This will take away from the simplicity of the design and make it more difficult to use. Lastly, the font does not change throughout the app, nor does it use Serif font for headers or San Serif for extended text.

The *Profile* page does not differentiate between the title and the actual piece of information. For instance, closing off *address:* and then having the user's address could be a more user friendly design, or even using a bold font type to separate the information accordingly.



Overall, the application does not display a visible feature for returning to the home screen and continuing on to another page. For instance, if a user has edited their profile and wanted to look at an upcoming trip, how do they navigate to the *Trips* page from the profile page? This is not a completely visible and shows that this design had hidden capabilities.

IV. Design 3



For design 3 we chose a horizontally tabbed method. This has tabs at the bottom of every screen that are the profile, trips, records, and about us sections. These allow to users to easily at any time switch to any other section of the application they wish to look at. On the login screen there is also emergency contact info as well as any other information the user wishes to put on the login screen. This can include medical conditions and any allergies that would allow a stranger to help the user if necessary.

Strengths

This follows the conceptual model that many IOS apps follow and will be familiar to the user. Common fate was taken strongly into account when designing the various screens of the app and they are gridded well. There are clear labels for what each screen is.

Proximity and similarity were taken into account when designing the login and profile screens. This allows the user to easily identify what each label is describing. Similarity helps the user distinguish between the various aspects of the trips screen such as different destinations. The about us screen matches the user's conceptual model of what an about us screen should contain.

Weaknesses

Some of the downsides of the horizontal tabbing include the possibility of accidentally changing screens. Another bigger issue is the fact that there is limited screen space on a phone, and having the tabs visible at all times could take up space on the user's screen. On all screens in the top right corner is Scrubs, and in the top left corner is the label of the screen. This could also take up space on the user's screen. All of these in combination could take up a decent portion of the user's screen. This could be an issue especially for the users who have phones with smaller screens.

V. Summary & Reflection

When we began this stage of the design we had one thought in mind, simplicity. We wanted our designs to be easy to navigate with the fewest possible pages all while being user-centered. While designing our interface, we realized we needed to incorporate many aspects due to the needs of our audience. For example, our application may sometimes be used in emergency situations, so speed and ease of use are of utmost importance. This idea combined with the topics we have learned in class allowed us to accomplish our goal. Consequently, we created 3 designs.

We didn't arrive at these designs without a little research, we began by looking at our phones and critiquing the most common apps we use on a daily basis. What we liked and disliked about each of them. For example, Snapchat implements an interactive swipe feature, while Sleep Cycle has a tabular menu at the bottom. The interactive swipe feature proved to be useful at saving space on the screen while the tabular menu was the quickest to change pages with. After learning more about the demographics of our audience, we know the interactive swipe interface will resonate more with younger users while the tabular menu will be more familiar for older users. After a while, we discovered a few common trends and decided to implement them with our own twist.

The design process didn't stop once we had determined the three layouts, we decided to focus on the small details that can make all the difference for a user. We began with the four Gestalt principles and applied our knowledge of these principles to the designs. The vertical hideaway menu and horizontal Tabular menu both use closure and common fate to help the user navigate the app. All three designs use similarity and common fate to create a typographical hierarchy to display information. This allows the user to quickly

and easily view groups of information that they can interact with. Other design principles were implemented such as alignment which help keep our information organized in a clean way so that the user is not confused by the placement of elements on the screen.

After creating our designs, we were able to get a much better idea about how our users might react to the different implementations we created. It is clear that we must accommodate a variety of people that are used to many different interfaces. Although our application has 4 simple core functions, they can easily be confusing to the user if not implemented correctly. Breaking down the strengths and weaknesses of each of our designs has helped us understand how to ensure that these 4 simple core functions are easily executed by the user.