**Prototypes**

**Chatfuel**

Chatfuel allows you to create a chatbot through Facebook Messenger.

It first requires a “Welcome message” that briefly explains the use and some of the functions of the chatbot. After the welcome message, you then have to specify some of the actions the user should be able to perform. Two main things that you can do are giving the user a short list of choices that they can choose as their response, or allow the user to type any response in that they want. Giving users a list of choices requires a different “Block” for each choice. The block is simply a way for the chatbot to respond to the answer choice that the user selected. The example shown is a chatbot for SIS that is supposed to give information about courses, grades, and finances. We could have added a couple buttons to give users these choices to select from, but in this case, we decided to allow the user to respond freely. This requires the use of the “AI” simulator. Here, you can specify some keywords or phrases. If the user types in one of these, the chatbot will respond with the appropriate block. Let’s try this out on Messenger. We can type a command that we want to do (viewing tuition for next semester, viewing grades, or searching classes that are offered). [ . . . ]

**Proto.io and Invision**

Proto.io and Invision allows you to create a simulation of a mobile app. Proto.io is used for creating the different screens in the app. The example shown is a mobile app for SIS. The text boxes, shapes, and images are all found in Proto.io’s libraries. They can easily be inserted onto the mobile screen by clicking and dragging what you want to add to your screen. Once you’ve created your screens, you should download all of the images. In Invision, import all the images. Invision is used for creating “hotspots” on each of your screens, so that you can add some functionality for changing from one screen to the next. You can choose which buttons you want to make active by creating a hotspot around the button. Then, you can choose whether you want to have the user tap, double tap, or swipe in a certain direction to activate that button. You must also select which screen you want that button to take the user as well as which transition you want to use (dissolve, slide, flip, etc.). Let’s see how these hotspots work on the screens. [ . . . ]

**Twine**

Twine allows you to create a simulation of a web app. It doesn’t allow for much design of the actual webpage, but it shows the flow of the web app and how the pages are hyperlinked together. The first step is to create a main page that the web app goes to first. Any text that you want to appear to the user is just written normally. This example also uses SIS, and “Welcome to SISbot!” would be the first message that would appear to the user. To add any buttons you want to allow the user to click on, you have to write the button name (that appears to the user) followed by an arrow and then the name of the “Passage”, which is the page that you want the user to be taken to, and then all of that has to be enclosed in double brackets. If the passage name exists, then the two passages will be linked as expected. If the passage name does not already exist, Twine will create one for you to remind you to include information in that passage so that you don’t have any dead links. Let’s take a look at the SIS example to see how the buttons work. [ . . . ]