

Final Sprint Log

Julie: 10, 11, 12

Evelyn: 7,9,13,15

Aolin: 5, 8, 14

Noah: 1-4

George: 6

User story #	User Story details	priority/ Importance
1	Add basic HTML code for sidebar.html, and add a drop-down menu for choosing resume template.	
2	Add a drop-down menu for choosing text color.	
3	Add all buttons that are needed, i.e. submit, buttons that open subsection dialogs. No functionality for these.	
4	Add a drop-down menu to choose sections or entire resume (?)	
5	Create an HTML file that will be the input for Resume Header with inputs for all required fields. (Name, title, social media accounts, etc)	
6	Create an HTML file that will be the input for Education with inputs for all required fields.	
7	Create an HTML file that will be the input for Experience with inputs for all required fields.	
8	Create an HTML file that will be the input for Skills with inputs for all required fields.	
9	Create a pop up dialog box to display any function that we planned to use or display anything required info in the project announcement(instructions for the user as well as explanation of program structure to the grader of this assignment)	
10	Document SCRUM meetings, take notes about progress. (Ongoing)	

11	Create two user stories.	
12	Design complete user study to be used in phase 3.	
13	Create report.docx and write instructions on the running project.	
14	Add initialization code in code.gs to display sidebar	
15	Add html file to include Survey question (Customer Feedback)	

Original Sprint

User story #	User Story details	priority/ Importance
1	All the drop down menus needed. One drop down menu for choosing resume template, one drop down menu for choosing text color. One drop down menu to select whether sections or the whole resume, or it might be changed to other things.(see below) All the regular buttons needed(buttons for submit, buttons for generate graphs,etc), currently in this phase, just make them print something or they don't need to link to classes or function yet,could be just printing in javascript console.	4
2	Everything in a survey. Think of 10 questions that let the user to self evaluate their resume. Build radio buttons(for each question, there are 5 radio buttons for the user to click, each radio button will be an integer, i.e. 1,2,3,4,5. The question will be displayed above the radio buttons). Display the 5 radio buttons each with a number. Radio buttons don't need to have any action in the background when user clicks them	3
3	All text entry fields using <input> tags <form> tags like we did in project one and one pop up window that contain text entry	2

	<p>fields for users to enter social media accounts, and buttons to link back to other places for mouse events. Find out how the google document works and set up the entire document like <code>getBody()</code> <code>getParagraphs()</code> etc.</p> <p>The drop down button for section/entire resume might be changed to one button with multiple text entries. Not sure about the implementation yet. Whoever works on part 1 need to communicate with the person working on part 3</p> <p>Create a pop up dialog box to display any function that we planned to use or display anything required info in the project announcement.</p>	
4	<p>Documentation of details of every meeting. Taking meeting notes and update notes every day right after the meeting.</p> <ul style="list-style-type: none"> • user_study_design.docx or pdf: Include user stories and user study design, based on your implemented UI. See section 2, "Human Computer Interaction Requirement", above). • sprint.docx or pdf : sprint meeting logs , initial sprint backlog and final spring backlog, initial burndown chart, final burndown chart. Updated product backlog. • report.docx or pdf : include instructions to install and run, and include screenshots of the implemented user interface. <p>Overall system testing, have git working if we decide to use git. Help with Debugging. Don't need to be familiar with javascript.</p>	1

Sprint Meeting Log.

Monday 10/26 12pm

Members present: Allen Yang, Julie Herrick, Evelyn Tang, Noah Miner

- <https://stackoverflow.com/questions/33174024/how-to-declare-a-custom-class-in-google-scripts>
- The app script can't do any image manipulation, offers a plotting library to build charts
- May have to transfer data between docs and sheets
- Make buttons have "score 1-5" and then can build category charts around how the user answers
 - If we want to keep track of info we will need to keep track of it in a google sheet
- Change graph to be rating skills so that we don't need to keep track of multiple users stats
 - Like this <https://i.stack.imgur.com/BAxRV.png>
- Will make separate files for everything we need to turn in then combine it for each submission
 - <https://www.agilealliance.org/glossary/backlog>
 - <https://www.agilealliance.org/glossary/sprint-backlog/>
- Scrum process
<https://www.crystalloids.com/hs-fs/hubfs/Scrum%20Process.jpg?width=1800&name=Scrum%20Process.jpg>
- Product backlog will be the culmination of all of our sprint backlogs
 - First week will basically be doing the HTML file for the user interface
 - Would be good to have everyone do research into using google scripts so that we have the background knowledge once we get into more of the coding heavy phases
- <https://developers.google.com/apps-script/guides/dialogs#page.html>
- Finish HTML file before the weekend then we can style it
- Have information we input into whatever UI class we have
- <https://chrome.google.com/webstore/detail/google-apps-script-github/lfcgcmkmjiliehflfhjopckgpelofo/related> using GitHub with google script editor
- CSS style guide: <https://developers.google.com/gsuite/add-ons/guides/css>
- We should copy the code to our own project and not change directly in the shared drive file. Leave that to be the final working code.

Tuesday 10/27 8am

Members present: Allen Yang, Julie Herrick, Evelyn Tang, George Lan, Noah Miner

- Clarification on goal 3
 - What will the functionality of the buttons be?
 - Doc wouldn't update until the submit button was pressed, so we don't have to worry about updating it until submit is pressed
- We like how easy pushing/pulling is in the doc

- For reading data from the document. If someone put in a written resume could we scan it and then adapt it to our formats
 - Could be difficult with different formats, if the user marks corresponding sections that could work
 - Have an unnamed abstract section to make it easier to adapt it
- Will need to make an html file for each popup
- Task 4:
 - Does this mean generate a section or entire resume?
 - Yes
- What are user stories in task 11
 - From the project description, basically different scenarios for how people will use the app
 - User design study is questions analyzing how people will interact with the project
- Task 9 is kind of what needs to be done with all other tasks in this phase to show what they would do
 - Could have a popup with instructions on how to use the program
 - Would also have the popups based on what should be happening after user interacts with certain elements of the UI

Day 10/28 2pm

Members present: Allen Yang, Julie Herrick, Evelyn Tang, George Lan, Noah Miner

- Allen said he finished his tasks
- Everyone is added on the github repo now
 - George making a github with non TAMU email should fix the issue
- Noah: working on sidebar HTML and getting drop down forms working
- George: Going to start looking today
- Julie: Looking more into how to make user stories and user story design. Want to see our layout before I start too much
- Evelyn: Did HTML file for experience and unsure if that should be javascript since rn using HTML form tags
 - Not sure if it should be switched from HTML tags
- How to run the add on
 - From script choose run
 - Choose a document to test on
 - Open document then go to add ons
 - Select our add on and hover over to click run
 - Sidebar should pop up
- For this phase we just need to have the UI show up and pop ups to say what each one does
- https://developers.google.com/gsuite/add-ons/editors/docs/quickstart/translate#translate_gs

- Don't merge from google drive because that seems to replace the code rather than merging
- Don't push from google scripts, just pull. Push from a local branch
- In order to prevent overwriting code, only push to main branch when we're on the call
 - Otherwise push into our own branch

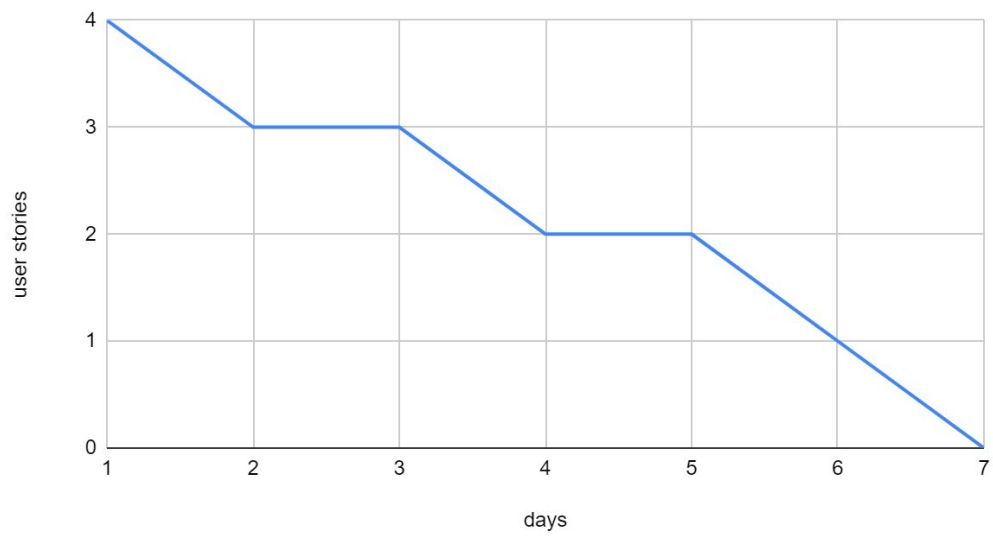
Day 10/29 8am

Members present: Allen Yang, Julie Herrick, Evelyn Tang, George Lan, Noah Miner

- Merging do it on the local branch so it doesn't mess up the main branch
 - Only use shared script to test so we don't want to mess it up
- Evelyn:
 - don't think inserting text box is possible
 - Section for entire resume isn't possible, can only insert into regular document
 - Use a table and insert info into different sections
 - Finished front end of task 7 (experience), looking more into the backend to prepare for the next phase
 - Still working on 9&13
- Julie:
 - Waiting until I see what the UI is looking more like in order to make sure the user stories are consistent with what we're doing
- Allen:
 - Done with tasks, willing to help if anyone needs it
- Noah:
 - Sidebar has all elements, working on style
 - Make sure functions are calling the popups Allen made
- George:
 - Working on education
 - Making a popup text box

Original chart

user stories vs. days



Final Chart

tasks vs. day

