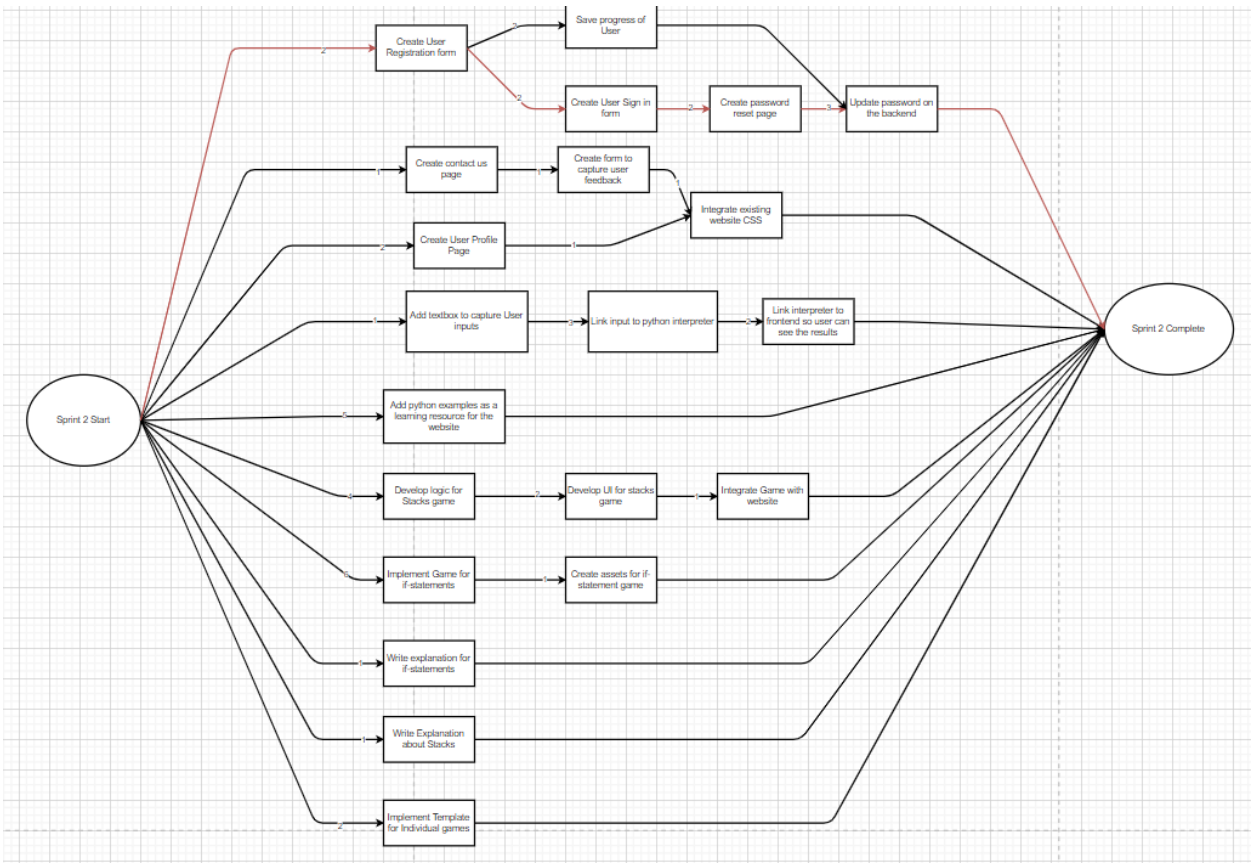
Network Diagram



Task	Time	Dependencies	CP
Create User Registration form	2		*
Create Contact Page	1		
Create User Profile Page	1		
Add textbox to capture User inputs	1		
Add python examples as a learning resource for the website	5		
Develop logic for Stacks Game	4		
Implement game for if-statements	6		
Write explanations for if-statements	1		
Write explanation about stacks	1		
Implement template for individual games	2		
Save progress of user	2	Create user registration form	
Create User Sign in form	2	Create user registration form	*
Create form to capture User feedback	1	Create Contact Page	
Link input to python interpreter	3	Add textbox to capture User inputs	
Develop UI for stacks game	2	Develop logic for Stacks Game	
Create assets for if-statement game	1	Implement game for if-statements	
Create password reset page	2	Create User Registration form, Create User Sign in form	*
Integrate existing website CSS	1	Create Contact Page, Create form to capture user feedback, Create User Profile Page	
Link interpreter to frontend so user can see the results	2	Link input to python interpreter, Add textbox to capture User inputs	
Integrate (Stacks) game with website	1	Develop UI for Stacks game, Develop logic for Stacks game	
Update password on the backend	3	Create password reset page, Create User Sign in form, Create User Registration form, Save progress of User	*

The network diagram and table on page 1 and page 2 represent the tasks for sprint 2 and each task's dependencies. The critical path for sprint 2 was the path involving DREAM-14 (account creation) and DREAM-16 (password reset) which contained 4 tasks.

To keep our sprint in schedule, we held daily meetings where each group member updated the team with their progress on their assigned user stories. This encouraged each group member to start their user stories early so they would have something to

present for each meeting. The meetings also doubled as help sessions where group members could discuss problems with user story implementation thus increasing the team’s efficiency.

At the end of sprint 2, we were unable to finish DREAM-14 (account creation) and DREAM-16 (password reset). The task “update the password on the backend” was significantly harder than we anticipated due the team’s lack of experience with frontend backend integration and databases. The web hosting service Heroku and the backend runtime environment NodeJS were both technologies that none of us had experience with. The group spent a lot of time figuring out how to interact with the Heroku Postgres database which left little time for us to implement the account registration and password reset features. What we learned from this experience is that we should account for the learning curve involved with using new technologies and adjust our sprint velocity accordingly.