Project Description  
Below are the user stories that you need to implement for the *entire* project. These are the full set of features, and **you are NOT asked to implement all of them for phase 1**. Features are explained in terms of user stories, in a layman's terms, and from an imaginary user's point of view. You should not expect the same level of detail as assignments. Details that are not discussed below are left out to you, and you get to design them. Your final grading will be done mostly based on the overall functionality of your website as if the stakeholder is testing your application.

**1on1**

* Users should be able to create accounts with basic information (name, email, password).
* Ensure that user data is protected and privacy is maintained.
* Users can create and manage a list of contacts (comprised of names and email addresses)
* Users can begin the process of scheduling meetings by creating a new calendar
  + On the new calendar, users should be able to specify their non-busy times which are available for meetings
  + Users should be able to indicate preference levels for different non-busy times on their schedules (e.g., high preference vs. low preference, or "book meeting at this time only if there is no other possible times that work" vs "definitely try to schedule someone here")
* Users can select contacts to invite to get added to this meeting calendar and a deadline for scheduling a regular meeting time
  + This should autogenerate an email to each contact, inviting them to schedule a regular meeting with the inviter
  + Email should contain a link to a Web page which lets a contact specify their schedule (times they are busy) and preferences amongst their non-busy time to meet with the person inviting them to schedule a regular meeting time
    - Each contact receives a unique link
    - Each unique link would allow the contact to open a Web page that knows who they are, who is the user that is inviting them to schedule a regular meeting time, and that user's calendar to try to add a meeting slot to
* Users should be able to log into the system to see how many of their contacts have provided the necessary information needed by the system to schedule the meeting slots
* Users should be able to remind users who haven't provided the information requested to do so
* Users should be able to see up to some number of suggested schedules determined by the Web site
* Users should be able to select a suggested schedule that they want to use and finalise it.
  + Upon finalising a schedule, an autogenerate email to each contact should be create to notify them about the time slot when they will meet with the inviting user.
* All users should be able to change their busy, non-busy time, and preferences until the calendar is finalised.
* Users should be able to move suggested meeting times with any user on any suggested schedule.
  + The Webpage should tell the user what times work and which times won't when they try to move a meeting time to a different time.

Requirements  
For phase 1, you are to implement most of the HTML/CSS for your project. Your final deliverable for this website is a **static demo** of 1on1. Your codes should only consist of HTML and CSS files, without implementing the backend or connections to servers from the frontend. You are optionally allowed to write some JavaScript code for this phase, or import JavaScript libraries from external sources. However, this is optional. Also, note that you will need to refactor your files when you migrate to React at phase 3. You can find an example of static demos [here](https://preview.colorlib.com/#summit). A static demo showcases the design of the pages by filling them up with some random test data. Forms and buttons do not send any request, but the links point to various pages included in the application.

Using a CSS framework (e.g., [Bootstrap](https://getbootstrap.com/), [Bulma](https://bulma.io/" \t "_blank), [Semantic](https://semantic-ui.com/), [W3CSS](https://www.w3schools.com/w3css/default.asp)

[Links to an external site.](https://www.w3schools.com/w3css/default.asp))  is strongly recommended as plain CSS will take enormous hours to implement and debug. However, since it is a static website at this stage, you cannot use any frontend framework, including any React-dependent CSS framework.

The detailed structure or the exact number of pages is up to you. However, your website should be navigable. That is, different pages should be connected to each other through clickable links and buttons. The set of static pages should demonstrate your thinking about how to handle different usage context, such as:

* Users having used the Website in the past to schedule meetings (i.e., more than 1 calendar and some of those calendars have been finalised already)
* What a finalised schedule looks like vs. what an unfinalised schedule looks like
* Schedules where not all of the contacts have supplied their information vs schedules where all of the contacts have supplied their information
* Different suggested schedules
* The ability for the user to move suggested meeting times for some contacts vs. when they aren't able to move the meeting times for some contacts
* Errors, such as incorrect email addresses supplied for some contacts

Extra Features  
You are required to come up with your own "extra feature" for the website, which will be graded during phase 3 of the project. It is highly suggested that you make a prototype of the extra feature page during this phase, but is it not required because we will not mark it until phase 3. Note that this extra feature is worth 5% of your overall grade, and thus it should be challenging (e.g., it can't be a bunch of static pages), and the amount of effort that you put into it should be reasonable.