Dasha Asienga, Priya Bakshi, and Lynca Kaminka.

Our design did not change much over the course of working on this project. One aspect that we decided to not implement was the syncing of multiple users into one calendar. While users are not able to perform this function, multiple users are able to use the app at the same time and on the same device because every user's login credentials are different and linked to their inputted tasks. With a bit more work, we may have been able to combine 2 different files for an existing account if the user wished so, and then load that onto the calendar upon a request to sync.

We had also planned to use different class variables for the 3 main events in our application (Activity, To-do Task, and Reminder), but we ended up not needing them because of the way we designed our program. We ended up relying on different files to store different information so we did not need a class holding information about each event.

Many parts of our original design were useful in our final project. One especially useful part of our original plan was to use LinkedLists to store the event names, event start/end times, and all the other relevant data for the events. This made it easier to write to file and also read to file by first storing information on the LinkedLists before populating the files. We found this especially helpful because we were able to add the names of events and tasks without setting a limit on how many a user could input. Another aspect of our project that we found useful was having multiple components to the app (the calendar, to-do list, and reminders). It allows for a more functional app that users can use for many different aspects rather than just having one use for the app. Our original design also included a Calendar Grid, which held information about the calendar layout, but the JPlanner package we used was able to replace that and provide us with a ready-made grid. We also intended to have a CalendarOwner class to store information about each user. While we didn't end up needing this class, LoginPage and IDsandPasswords classes do exactly what we had envisioned CalendarOwner to do.

While we ended up having different classes, a lot of them were rooted in the ideas we had at first but were modified due to the nature of the programming process, which enabled us to make changes as we go.