Integration Strategy

For our project we integrated code by building on different components simultaneously. We started by creating an interface and buttons then we added functionality to those buttons. Finally, we tied in the quizzes that we had made. We were also developing new functionality using React. We are joining the functionality of the react app to our site once it is completed. Because we are meeting in the middle, this is a sandwich integration. It is a sandwich integration because we are implementing logic artifacts top down and implementing operational artifacts bottom-up.

Ideally our integration would be sandwich integration so that we can discover faults sooner and easier. Sandwich integration allows for an adequate amount of testing and major faults would show up sooner. Although we had characteristics of Sandwich Integration, we also had characteristics of all-at-once integration, since we are joining components and integrating them together late in development. All at once integration is not the best integration strategy since major faults show up later and there isn’t as much time for adequate testing. For our project we had characteristics of both sandwich integration and all at once integration, since we worked top down on logical artifacts and we worked bottom up on operational artifacts however we merged artifacts later in the process, so it could be somewhat considered an all at once integration. Realistically we are not getting all of the benefits of sandwich integration since we are combining artifacts later than most integrations, however we are still getting a good amount of benefit from attacking the project from both sides.