EECS Team 10
Integration Strategy
11-5-20

Our team used Top-Down Integration strategy, in order to integrate code from each other members. We started coding each individual component and slowly build the next step from that one step. For example, the database we integrated from one of our members is later passed with a key to other members to test and experience with. After finishing experimenting, we will build code on top of that to see if there are any issues. Then we will continue development. A weakness of Top-Down is the fact that if something is amiss then we will need to make sure that it's not our small components we integrated that isn't working due to the fact that we don't rigorously test that component as a whole. One good example is that we are assuming that the message is correctly send to every single other connection aside from the one sending the text. After that components works, we will try to add a new feature like blocking and muting. The issue is if the actually sending isn't working then implementing blocking and muting won't work due to the fact that it works on the fluke and we didn't do enough testing. The current process of integration is we work on the component we need to be coded then after that we start building of that component like database storage then database retrieval right after to test with chatlogs. We will send the working database storage and retrieval component to the necessary groups to work on their feature and they will add it in their code base while testing their own code with it.