

Retrospective

Main Criteria

- Has the retrospective identified genuine problems with the team's policies and practices?
 - Yes, the retrospective identified multiple problems that the team will strive to solve in the next sprint.
- What strategies has the team decided on to address any identified problems?
 - These strategies include but are not limited to not crowding the group chat and using direct messages, communicating better about the work you are doing or are supposed to do, finalizing the scope earlier, and more.
- Are those strategies likely to work?
 - Yes, because we have all shared our thoughts and opinions, so the strategies that we have decided on have been peer-reviewed and consist of our collective thoughts as a group.

What went well?	<ol style="list-style-type: none">1. Better burndown progress compared to the last time2. Improvement in programming skills3. Everyone completed the assigned tasks according to their own roles.4. Team collaboration in sharing work and completing everything in the end.5. Each member is more familiar with the workflow, and the entire sprint is more manageable.6. Got better at following the git policy in terms of branching.7. We started coding immediately after our sprint meeting.
What could have been done better?	<ol style="list-style-type: none">1. Start work earlier2. Check burndown chart more frequently3. Communication in terms of work done4. Confirm the project requirements and user stories with the client as early as possible to avoid wasting time on redoing or correcting.5. Git commit messages, since we were not aware of the actual format until we received feedback for the last assignment.6. Do documentation simultaneously so that we have time to review

What will we try next?	<ol style="list-style-type: none">1. Do the client interview earlier and confirm scope of the sprint earlier to ensure we are at the right pace and path.2. Allocate the tasks better and earlier3. Communicate more in the group4. Communicate more with the client.5. Explain in details what to do for each tasks6. Better private message for individual corrections so that the group is not crowded.
What questions do we have?	<ol style="list-style-type: none">1. Would it be better if we pair front-end and back-end developers to ensure a fair distribution of workload?2. Should we ask the client for the scope of sprint 3 and the actual MVP requirements on the day of the demo itself?