Monochrome is a top-down adventure-puzzle game where players can move around to position themselves, move their shield to bounce paintballs, which are shot by the player's or another cannon. Guide the paint balls, in the colorless world left behind by Monochrome's devastating attack, and find its weakness, in order to restore color, and life to the town.

From the start we all agreed on the puzzle genre for our games. We looked around for inspiration and we were very intrigued by the design and gameplay mechanics of *Celeste* and *Portal*. Both games possess very simple core mechanics, and the puzzle designs and "twists" to those core mechanics essentially add depth to the games. In Chapter 5 of *Celeste*, the player has to carry a character Theo -- who is trapped inside a crystal and thus can be used as a damaging projectile -- through the levels, which is similar to how players oftentimes need to move around boxes to trigger buttons in *Portal*. This gave us the initial idea for our game's core mechanic: Players can shoot projectiles to bounce off objects, activate triggers, and advance to the next levels.

Our intended player impact for this mechanic was to create a puzzle game that also required a lot of precision and skill from the player. The requirement to shoot and position your character as well as move the shield into the correct angle to bounce the ball made the game mechanically challenging, which we hoped for.

With our idea in mind, we created a gold spike build in 2 weeks to prove our idea was possible. Our gold spike build was 1 level with a puzzle that required the player to shoot in a fire zone, bounce off walls, bounce off their own shield, or bounce on drag and droppable keys. What we had after this deadline was a unique puzzle game that required the player to not only solve the puzzles given to them, but also have the mechanical skills to execute their solution.

When we received playtesting feedback from our peers and instructors, we found that one of our core mechanics, the drag and droppable keys to form bounceable walls, was not being used. The players either did not know that the mechanic existed, or felt no need to use it. Based on that feedback, we removed that mechanic and focused our attention on the other

parts of our core gameplay. We also iterated on the structure of our game and turned our game from a series of levels to an open hub world with branching sections of levels. There were a lot of additional things in our game that we iterated on based on weekly feedback, such as the art style, introduction, and many many bug fixes.

Every week or two weeks on some occasions, our team would receive video feedback of peers and instructors playing our game. These videos were think-alouds, where the testers would just say what comes to mind while playing the game. This gave us a lot of good information on current problems in our game, how the game could look in the future, and current things that we are doing well with.

A lot of things went well during this project. We were able to communicate openly about what we wanted the game to ultimately become. Listening to each others' feedback such as positive and constructive information. Without these kinds of clarifications, we could potentially have run into delays due to disagreements, or certain members not being comfortable with a certain design, without being able to speak their minds. Focusing on working as a group, we tried our best to divide responsibilities evenly, and when finished with tasks, actively asking if anyone needs assistance. Our team also came together to problem solve at any moment when it was needed, and if not everyone was available, at least two of us seemed to always be addressing any problems that came up. It was great to see that each member would take point on problems, and have support at most times, and with these habits, we did not run into delays with certain problems being avoided, or just not voiced at all.

However, there are some things that went not as well as we hoped for during the project. One issue we had was with git and merge issues. At some point in our project, we lost a lot of progress because of a merge conflict we were unable to fix, and ended up having to move to another branch to continue to work. I think in future creative endeavors, this can be avoided by making committing, pushing, and pulling a more frequent habit. Also clearer communication on the current sections of the game we are working in order to avoid merge conflicts. Another thing

that we hope to improve on is the overload of work near the end of deadlines. Many times we found ourselves working late at night a day or two before the deadline, which left us little room to have errors. In the future we hope to manage our time more evenly and start work earlier.