- Regarding the game proposal document, I wrote the “Brief Statement of Game Concept,” “Target Platform,” “Development Tools to be Used,” and “Group Coordination Plan” sections.
- Also helped edit the game proposal document
- With the exception of the gun pixel art (the pistol, shotgun, and assault rifle), created artwork for the entire game, using photoshop and UI design inside Unity, including:
    - All UI, including the main menu, store UI, other main menu subpages, and everything on the in-game UI overlay
    - All sprites, including sprite sheets for animation, and hallway chunks
- Implemented player movement
- Made the camera follow the player (through a script, not by making it a child)
- Made the player rotate to look at the mouse
- Implemented the bullet prefab (was later split into 2 different prefabs: playerBullet and guardBullet)
    - playerBullet and guardBullet are copies of the original bullet prefab with minor changes specific to the player/guard
- Implemented all weapon shooting, bullet instantiation, and bullet collision
    - Including the pistol, slingshot, shotgun, and assault rifle
        - Including the rock prefab that the slingshot shoots, but I did not implement the sound the rock makes
- Implemented the health bar for the player
    - Health bar for guards are copies of the health bar for the player, with minor changes to allow it to work with instantiated guards
- Implemented takeDmg function for the player, as well as the player’s health
- Set up a lot of the ignore layer collision parts in many gameobject’s scripts, including player bullets avoiding the player, other player bullets, gold coins avoiding guards and bullets, etc
- Fixed many bugs, including guard bullets not damaging the monkey, max value of guard health bars weren’t correct, an issue where bullets were passing through walls without colliding, errors caused by some variables being static previously (made them not static anymore), including the changes in other scripts required (since the variables weren’t static anymore), added default values to some PlayerPref retrievals, guard’s health bar not being destroyed when the guard died, …
    - Note: I did not mention some bugs associated with implementations/features solely implemented by myself
- Implemented the entirety of level generation, including:
    - Generating a continuous path of hallway chunks
    - Spawning guards, including spawning patrol points for the guards to patrol between
    - Spawning clusters of gold coins

    - Made the frequency of guard spawns depend on the floor number (more difficult the farther the player progres
- Implemented the entirety of the menu scene (both art and functionality), including the store UI (this took an especially long time), main menu screen, How To Play screen, Credits screen, and options screen, resetting the save file to a new game (wiping progress), with the exception of:
    - Making the price of upgrades increase the more that upgrade is purchased, and loading that information from the save file when loading the store (to display the price tags)
    - Adding menu music
    - The content written in the “How To Play” screen (although I did move this content from a Google Doc into Unity)
    - The content written in the “Credits” screen (although again, I was the one who actually added the content into Unity)
- Implemented saving the data from the store UI into the save file (PlayerPrefs)
- Implemented loading data from the save file to set the store UI to the correct state (showing what is purchased, and what isn’t)
- Implemented loading data from the save file to set the parameters in the actual game (monkey health, speed, weapon damage, etc)
- Edited the script for the Video Demo
- Got gameplay footage for the Video Demo
- Edited the Video Demo
    - Edited audio placement
    - Edited gameplay footage
- Worked on guard prefabs and AI scripts, including:
    - Only having guards shoot at the monkey if they are a certain distance from the monkey
    - Implemented guard aimbot (making guards aim at the monkey before they shoot)
    - Implemented setIntriguePoint function (based off previously written code) for the purpose of generalizing the purpose of this function
- Changed the gold and distance tracker UI to scale with the resolution of the screen
- Implemented the floor indicator UI (top left corner of screen)
- Implemented the ammo tracker UI (bottom left corner of screen)
    - And functionality of ammo, not allowing player to shoot if they’ve ran out of bullets
- Implemented the hotbar, including indicating which weapon was selected (bottom middle of screen)
- Implemented toggling between sprinting and sneaking

    - Implemented both sneaking when holding shift, and using shift to toggle between sprinting and sneaking, depending on the setting to enable/disable toggle for sneak/sprint
- Implemented gaining gold based on distance traveled
- Conducted all playtesting surveys and recorded feedback from all playtesters
- Editing/quality-control for playtesting and progress report
- Implemented death screen and level completed screen (including functionality, and when they are triggered)
    - Including restart button, which resets all elements of the scene, as if the scene had just been loaded
    - Made the floor tracker increment by +1
- Scaled volume on all audio sources, as 100% volume was way too loud, and 60% of the volume (as the actual Unity volume) was sufficient for the maximum volume (scaled to 100% of player controlled volume)
    - 60% is just an example, each audio source was scaled differently
- Conducted the majority of balance testing
- Edited the Final Video Script
- Editing/quality-control for the Wrap-Up Report
- Got gameplay footage for the Final Video
- Edited the Final Video