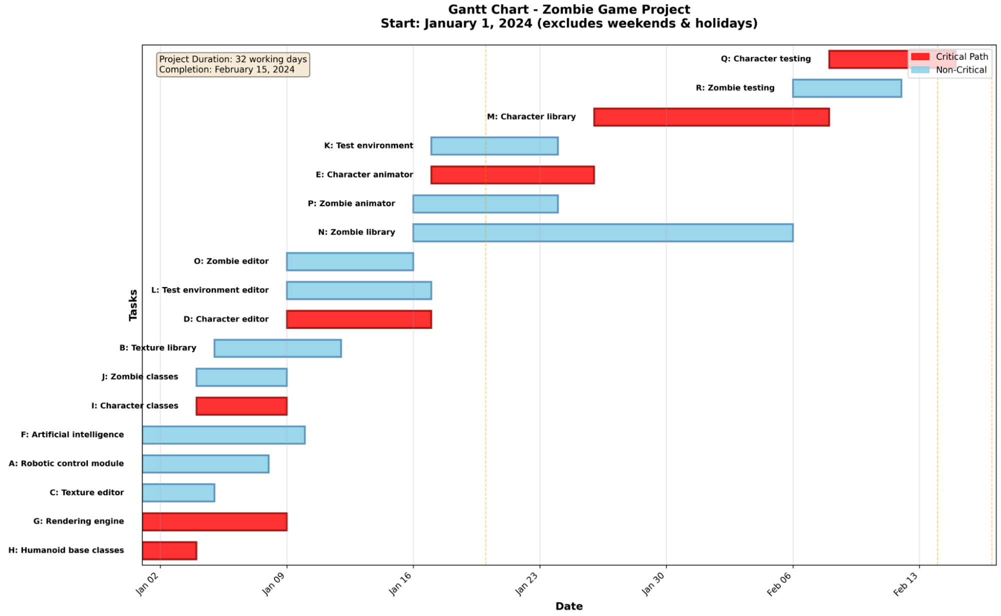
1. The basic tasks that all software engineers must handle are finding the requirements, the high and low level design, the development and testing, and finally deploying and maintaining.
2. -
    a. Requirement gathering is figuring out what the customer wants and needs, and writing that down, so the developers know what needs to be done.
    b. High-level design, breaking the project down into the big pieces, explaining how the big components will work with eachother
    c. Low-level design is creating detailed designs for each part of the project
    d. Development actually involves writing the code that the project will use.
    e. Testing is where the code is tested to make sure that it actually works asnd does what it was sent out to accomplish
    f. Deployment, releasing the software to the users, which can include things like installations or training data.
    g. Maintenance, fixing bugs that are discovered or adding new features.
3. Some things that I noticed show the different changes that were made over time, it also shows who made the changes. This is similar to GitHub, because they are both forms of version control, but Google Docs auto saves and dosen't need commit messages, while with Github, you need to "save" manually with commits and include a message to explain what was done.
4. JBGE stands for Just Good Enough, it means that when creating documentation, you should be just writing the bare minimum to understand the code. Some reasons for that are that writing long documentation takes time, the upkeep for it is more tedious if it is too long.
5. -
    a. The total time for the tasks using the critical path method ES, EF, LS, LF for all 18 tasks
    b. The critical path is G > H > I > D > E > M > Q with no slack
    c. The project will take 32 working days
6.



7. You can’t plan for unpredictable problems, but you can include extra time in your plan to account for something going wrong. You can include buffer time, have a plan B ready for when something goes wrong, and pay attention to the development process so you can jump into action when something goes wrong.

8. 2 of the biggest mistakes you can run into when tracking tasks are not taking action once something starts to go wrong, because ignoring the problem will cause it to grow into a much larger problem down the line. And the other mistake is throwing more people at the problem, because some people will think that more manpower will solve the problem faster, but it usually just leads to more confusion.
9. 5 characteristics of good requirements
    a. Clear- easy to understand
    b. Unambiguous - has one interpretation and won't get misinterpreted
    c. Consistent - dosent’ contradict other requirements
    d. Prioritized - requirements are shown by importance
    e. Verifiable - you are able to test if the requirements have been met or not
10. -
    a. Allow users to monitor uploads/downloads while away from the office. Category: B (Business)
    b. Let the user specify website log-in parameters such as an Internet address, a port, a username, and a password. Category: U, F (User, Functional)
    c. . Let the user specify upload/download parameters such as number of retries if there's a problem. Category: U, F (User, Functional)

d. Let the user select an Internet location, a local file, and a time to perform the upload/download. Category: U, F (User, Functional)
e. Let the user schedule uploads/downloads at any time. Category: N (Nonfunctional)
f. Allow uploads/downloads to run at any time. Category: N (Nonfunctional)
g. Make uploads/downloads transfer at least 8 Mbps. Category: N (Nonfunctional)
h. . Run uploads/downloads sequentially. Two cannot run at the same time. Category: N (Nonfunctional)
i. If an upload/download is scheduled for a time when another is in progress, it waits until the other one finishes. Category: N (Nonfunctional)
j. Perform scheduled uploads/downloads. Category: F (Functional)
k. Keep a log of all attempted uploads/downloads and whether they succeeded. Category: F (Functional)
l. Let the user empty the log. Category: U, F (User, Functional)
m. Display reports of upload/download attempts. Category: U, F (User, Functional)
n. Let the user view the log reports on a remote device such as a phone. Category: U, F (User, Functional)
o. Send an e-mail to an administrator if an upload/download fails more than the maximum retry number of times. Category: F (Functional)
p. Send a text message to an administrator if an upload/download fails more than its maximum retry number of times. Category: F (Functional)
The one category missing was implementation

5.9

Must have
- Difficulty levels
- Way to keep score

Should have
- Hint system
- Sound
- Achivments

Could have
- Daily challenge
- Animations
- Leaderboards

Won’t have

- Custom word list
- Multiplayer