HW 1

Problem 1.1

Requirements definition, design, implementation, testing, and maintenance.

Problem 1.2

Requirements - determining what the software must do and the constraints under which it must operate.

Design - planning the structure of the system, including architecture, components, and interfaces

Implementation - writing the actual code that fulfills the design and requirements

Testing - verifying that the software works correctly and meets the specified requirements

Maintenance - updating and fixing the software after deployment to correct defects or add enhancements

Problem 2.4

A timestamp and version name, user who made the changes, color coded highlights indicating additions, deletions, and edits

Differences between versions are displayed inline in the document, allowing users to visually track what text was added, removed, or modified. Users can also restore previous versions

Comparison to GitHub:

Similarities:

Both track historical versions of a file, both allow viewing differences between versions, both allow restoring earlier versions

Differences:

Google Docs focuses on document editing with visual highlights, while GitHub uses line-by-line diffs, GitHub supports branching, merging, and commit messages, and GitHub is designed for source code collaboration, while Google Docs targets general documents

Problem 2.5

Just Barely Good Enough, meaning software should meet requirements adequately without unnecessary features or over-engineering

Problem 4.2

Task	Earliest Finish
A	5
B	14
C	4
D	15
E	22
F	7
G	6
H	3
I	6
J	6
K	21
L	16
M	31
N	36
O	16
P	22
Q	35
R	40

Critical path is:

C → B → N → R

Total Expected Project Duration

40 working days

Problem 4.4

Gantt Chart for the Critical Path

Start Date: Wednesday, January 1, 2024

Task	Start Date	End Date
C	Jan 2	Jan 7
B	Jan 8	Jan 14
N	Jan 15	Feb 4
R	Feb 5	Feb 10

Problem 4.6

Projects can handle unpredictable problems by building schedule buffers, managing risks proactively, maintaining flexible plans, communicating frequently with stakeholders, and being willing to re-prioritize or adjust scope when unexpected events occur

Problem 4.8

Failing to update task status regularly and tracking tasks at too coarse or too fine a level of detail

Problem 5.1

Clear and unambiguous, complete, consistent, testable, feasible

Problem 5.3

Requirement	Category
a	User
b	User
c	User
d	User
e	User
f	System

g	System
h	System
i	System
j	System
k	System
l	User
m	User
n	User
o	System
p	System

Yes, both user oriented and system oriented requirements are present.

Problem 5.9

Must have random word selection, letter disabling after selection, win/lose messages

Should have difficulty levels and visual progress indicator

Could have sound effects and hint system

Won’t have multiplayer mode and online leaderboards