Exercise 1: Communication & Role Clarity

Team Roles, Responsibilities & Communication Plan

Role	Main Responsibilities	Key Communication Channels	Communication Frequency
Producer / Project Manager	• Plan schedule and milestones • Assign tasks • Track progress & risks • Coordinate the team	• **Meetings**, **Chat**, Documentation	• Daily check-in
		• **Chat**, **Documentation**, **Meetings**	• *Weekly planning* meeting
Game Designer	• Design gameplay mechanics • Balance systems • Write game design documents (GDD)	• **Chat**, Documentation, **Meetings**	• Daily chat updates
		• **Chat**, **Documentation**, **Meetings**	• Weekly design review
Programmer / Developer	• Implement gameplay systems • Fix bugs • Optimize performance	• **Chat**, **Version** Control comments, **Meetings**	• Daily chat updates
			• Weekly tech sync
Artist (2D / 3D)	• Create characters, environments, UI • Maintain visual consistency	• **Chat**, Asset Repository, Documentation	*Daily progress updates*
Sound Designer (*Optional*)	• Create music & sound effects • *Integrate audio feedback*	• **Chat**, **File Sharing**	• Weekly updates
QA / Tester	• Test builds • Report bugs • Verify fixes	• **Bug Tracker**, **Chat**, Documentation	• After each build
			• Weekly reports

Communication Channels

Channel	Purpose
Chat (Discord / Slack)	*Daily communication, quick questions, progress updates*
Meetings (Voice / Video)	*Planning, reviews, problem-solving*
Documentation (Notion / Google Docs)	*Design documents, rules, decisions*
Version Control (Git / GitHub)	*Code collaboration, change tracking*
Task Board (Trello / Jira)	*Task assignment and progress tracking*

Weekly Communication Plan

Daily (10–15 min.):

- Each member shares:
  - What they did yesterday
  - What they'll do today
  - Any blockers

Weekly (30–60 min.):

- Review progress vs goals
- Playtest latest build
- Adjust priorities
- Clarify responsibilities if needed

- Reduces confusion
- Improves accountability
- Keeps everyone aligned
- Prevents communication overload

Exercise 2: Feedback & Iteration

Objective Implement a structured feedback process to improve the prototype while keeping communication constructive, objective, and actionable.

1. Feedback Framework

Method: **Observation → Impact → Suggestion**

This framework ensures feedback is clear, respectful, and focused in imprrovementt rather than personal opinion.

Step	Description
Observation	Describe what you saw or experienced in the prototype (facts only).
Impact	Explain how this affects gameplay, usability, or player experience.
Suggestion	Propose a concrete and actionable improvement.

2. Example of Constructive Feedback

Feature: **Player movement speed in the prototype**

- **Observation:** The player character moves very quickly in small rooms.
- **Impact:** This makes precise movement difficult and causes players to to collide with walls, reducing control and comfort.
- **Suggestion**: Reduce base movement speed by 15% or add a walk/run toggle for better control in tight spaces.

3. How Feedback Is Received & Applied

1. Feedback is collected during:
   - Playtests
   - Weekly review meetings
   - Written feedback forms or documents
2. All feedback is documented in a shared tool (Notion, Google Docs, or task board).
3. The team reviews feedback together and:
   - Identifies recurring issues
   - Prioritizes feedback based on impact
4. Approved feedback is turned into tasks and assigned to the appropriate role.
5. Changes are tested again in the next iteration

Exercise 3: Stress Management & Conflict Resolution

Objective Handle stress and resolve conflicts effectively to maintain collaboration, productivity, and team well-being during development.

1. Main Sources of Stress & Solutions

Source 1: Tight Deadlines

Description: Short timelines and milestone pressure can anxiety, rushed work, and burnout.

Solutions:

1. **Break large** tasks into smaller, manageable milestones with realistic deadlines.
2. **Prioritize tasks** clearly so the team focuses on what matters most first.

Source 2: Unclear Responsibilities

Description: When roles are not clearly defined, tasks may overlap or be neglected, creating frustration.

1. **Clearly define** roles and responsibilities at the start of each sprint.
2. **Use a task board** to show ownership of each task and track progress.

Source 3: Frequent Design Changes

Constant changes to features or direction càn demotivate the team & ìncrease stress.

1. **Limit major** design changes to specific review moments (e.g, weekly reviews)
2. **Document decisions** clearly so changes are justified and understood by everyone.

2. Conflict Scenario Description

Task Conflict

The game designer wants to complex combat mechanic to improve depth, while the programmer argues that it will take too long to implement and may delay the project.

This disagreement creates tension and slows down progress.

3. Conflict Resolution Strategy

Conflict Type

Task Conflict

The conflict is related to what should be implemented, not personal issues.

3. Conflict Resolution Description

- Effective leafeship-essential during stress and conflict situations; ns.
  - Stay neutral and listen to all sides
  - Encourage respectful, fact-based discussion
  - Focus on project goals rather than personal opinions.
  - Support compromise and collaboration

Steps to Resolve the Conflict

1. **Acknowledge the conflict:**
   - The team lead recognizes both perspectives and creates space for discussion.
2. **Clarify goals and constraints**
   - Review project priorities, deadlines, and technical limitations.
3. **Explore alternatives**
   - Discuss simpified versions of the mechanic or a prototype-first approach.
4. **Make a shared decision**
   - Choose a solution that balances design quality and development feasibility.