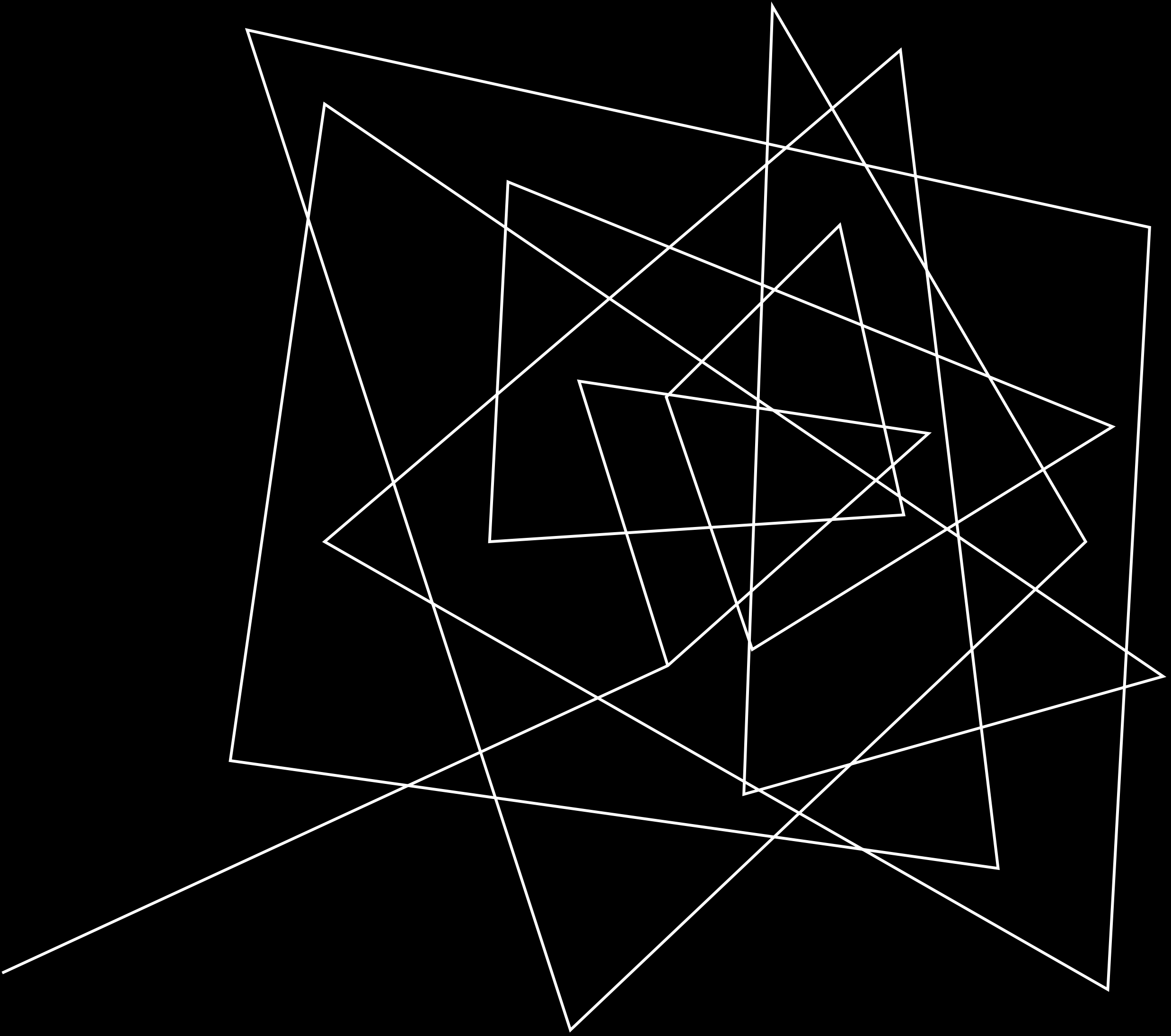

WRITING EXCELLENT SOFTWARE REQUIREMENTS

GROUP MEMBERS:

M.Saleh

AsadUllah Khan

Abbas Mushtaq

Zaighum

Abdullah

INTRODUCTION

Introduction

- **Importance of high-quality requirements:** Good requirements help ensure the final product meets stakeholders' needs.

- **Objectives of the presentation:** Learn how to write clear, effective requirements.

CHARACTERISTIC OF EXCELLENT REQUIREMENTS

Key Characteristics

- **Complete:** All necessary information is included.
- **Correct:** Accurately describe what is needed.
- **Feasible:** Can be implemented within constraints.
- **Necessary:** Must fulfill real needs.

COMPLETE REQUIREMENTS

Complete

- **Provide all necessary information:** Ensure no important details are missing.
- **Flag gaps (TBD - To Be Determined):** Identify any unknowns early.
- **Resolve TBDs before development:** Ensure all details are clarified before starting work.

CORRECT REQUIREMENTS

Correct

•**Accurately describe functionalities:** Ensure the requirements match what stakeholders need.

•**Verification with requirement sources:** Confirm details with those who provided the requirements.

•**Alignment with system requirements:** Make sure they fit with overall system goals.

FEASIBLE REQUIREMENTS

Implementable within system capabilities:

Ensure the system can support the requirements.

Consider project constraints:

Consider time, budget, and resources.

Technological limitations:

Be aware of what technology can and cannot do.

NECESSARY REQUIREMENTS

Necessary

- **Fulfill stakeholder needs:** Each requirement should meet a specific need.
- **Contribute business value:** Ensure it adds value to the project.
- **Compliance with standards and regulations:** Follow any relevant laws or standards.

WRITING REQUIREMENTS FROM THE RIGHT PERSPECTIVE

System perspective:

Describe what the system should do.

User perspective:

Describe what the user should experience.

CLEAR AND CONCISE WRITING

Writing Style

•**Clear and concise:** Use simple language and be brief.

•**Avoid ambiguity:** Ensure there is only one possible interpretation.

•**Use consistent terminology:** Always use the same terms for the same things.

APPROPRIATE LEVEL OF DETAIL

Detail Level

•**Detailed enough to avoid misunderstandings:** Include enough information to prevent errors.

•**Not overly burdensome:** Don’t include unnecessary details.

•**Depends on project complexity:** Adjust detail level based on how complex the project is.

USING VISUAL AIDS

Representation Techniques

- **Diagrams and models:** Use visual tools to explain requirements.
- **Data flow diagrams:** Show how data moves through the system.
- **State-transition diagrams:** Show how the system changes states.
- **Use case diagrams:** Show how users interact with the system.

AVOIDING AMBIGUITY

Avoid Ambiguity

- **Be specific:** Use precise language.
- **Avoid vague terms:** Don’t use words that can be interpreted in multiple ways.
- **Ensure shared understanding:** Make sure everyone interprets the requirement the same way.

ENSURING COMPLETENESS

Avoid Incompleteness

•**Comprehensive coverage:** Make sure all aspects are covered.

•**Regular reviews:** Continuously check requirements for completeness.

•**Stakeholder feedback:** Get input from stakeholders to identify missing parts.

SAMPLE REQUIREMENT - BEFORE

Example: Flawed Requirement

•**Example of a poorly written requirement:** "The system should be user-friendly."

•**Highlight issues (ambiguity, lack of detail, etc.):** Too vague, not specific enough.

SAMPLE REQUIREMENT - AFTER

Example: Improved Requirement

•**Revised version of the requirement:** "The system should allow users to complete the login process within 2 steps."

•**Explain improvements made:** Specific and clear, easy to understand.

COMMON ISSUES IN REQUIREMENTS

Common Issues

- **Ambiguity:** Requirements are unclear.
- **Incompleteness:** Important details are missing.
- **Misalignment with stakeholder needs:** Requirements don’t match what stakeholders want.

STRATEGIES FOR EFFECTIVE REQUIREMENTS

Effective Strategies

•**Involve stakeholders early:** Get input from those who will use the system.

•**Regularly review and update requirements:** Keep requirements up to date.

•**Use clear, precise language:** Avoid misunderstandings.

IMPORTANCE OF VERIFICATION

Verification

- **Ensure accuracy:** Check that requirements are correct.
- **Align with project goals:** Make sure requirements support the project’s objectives.
- **Validate with stakeholders:** Confirm that requirements meet stakeholder needs.

THANK YOU!