Software Quality Assurance (SQA) – Reviews

**What is a Review?**

*"A process or meeting during which a work product, or a set of work products, is presented to project personnel, managers, users, or other interested parties for comment or approval."*  
**[IEEE Std. 610.12-1990]**

**✅ Definition:**

A **review** is a formal or informal meeting conducted to examine a **software work product** (e.g., requirements, design, code, or test plan). The goal is to find **defects**, gain **feedback**, and **improve** the quality of the product.

**✅ Examples of Work Products:**

* Software requirement specifications (SRS)
* Design documents
* Source code
* Test cases
* User manuals

**✅ Purpose:**

* Ensure the **quality** and **completeness** of deliverables.
* Detect **errors early** (before coding or deployment).
* Gain **approval** from relevant stakeholders.

**Objectives of Reviews**

**🔸 General Objectives:**

1. **Identify required improvements** in the product.
2. **Assure completeness** of the deliverable.
3. **Ensure technical correctness**.
4. **Measure project progress** using the status of deliverables.

**🔸 Why are these important?**

* Reduces **development cost** by catching defects early.
* Prevents **rework** in later stages.
* Validates that the product is ready for **next phase** or release.

**More Specific Objectives**

**🔹 Early Defect Detection**

* Reviews allow early **identification of bugs**, which saves **time** and **money**.

**🔹 Deliverable Readiness**

* A reviewed and approved deliverable serves as a **foundation** for further development.

**🔹 Quality Assurance Before Development Continuation**

* Reviews **gatekeep** the transition to the next development phase.

**Kinds of Reviews**

**1. Business Reviews**

* Focus on **alignment** of the product with business needs.
* Ensure deliverables are **complete, correct**, and ready for the next phase.

**2. Technical Reviews**

* Performed by **technical team** to examine design, code, or architecture.
* Aim to **identify defects**, confirm working parts, and suggest improvements.

**3. Management Reviews**

* Check **project progress** and **readiness for management decisions**.
* Validate alignment with **project plan** and **budget**.

**4. Walkthroughs**

* **Informal reviews** where the author explains the product.
* Participants provide **feedback and suggestions**.
* Used more for **knowledge sharing** than correction.

**5. Inspections**

* **Formal process** with defined roles and steps.
* A **structured approach** for detecting defects.
* Focused on **preventing defect leakage** into later stages.

**Objectives by Review Type**

**✅ Business Reviews:**

* Ensure the deliverable is:
  + Complete
  + Technically correct
  + Ready for the next phase

**✅ Technical Reviews:**

* Identify areas for **improvement**
* Confirm areas that **don’t need change**
* Serve as a **filter** before moving forward
* Help detect errors that the original author may miss

**✅ Management Reviews:**

* Confirm project is on track
* Make **go/no-go** decisions
* **Resolve issues** needing management intervention

**Review Roles**

**In any structured review, the following roles are defined:**

| **Role** | **Description** |
| --- | --- |
| **Facilitator** | Manages the review process and keeps discussions focused |
| **Author** | Prepares the document and explains it during the review |
| **Recorder** | Logs defects and decisions made |
| **Reviewer** | Reads the material in advance and identifies defects or improvements |
| **Observer** | A learner or new member who watches the process |

**Responsibilities of Each Role**

**🔹 Facilitator**

* Provides background information
* Assigns roles
* Keeps the meeting on track
* Helps achieve consensus

**🔹 Author**

* Prepares the material for review
* Explains the document section-by-section during the meeting
* Clarifies issues
* Fixes problems found

**🔹 Recorder**

* Logs **defects** during the meeting
* Maintains **issue list**
* Records **decisions** and **tracks resolutions**

**🔹 Reviewer**

* Prepares ahead by reading the document
* Notes **defects and strengths**
* Confirms **rework** completion later

**🔹 Observer**

* A **trainee** or **new team member**
* Gains exposure to review techniques and project artifacts

**Review Guidelines**

**📚 Review Guidelines (Slide 19)**

1. **Preparation:** Review materials beforehand.
2. **Discussions:** Stay on topic; discuss only relevant issues.
3. **Respect:** Maintain professionalism and constructive tone.
4. **Agenda:** Follow the planned flow of the review.
5. **Review Records:** Keep written documentation of decisions and action items.
6. **Resources:** Ensure availability of necessary documents/tools.
7. **Attendees:** Right stakeholders must be present and engaged.

**Review Frequency**

Reviews should be performed at:

* Beginning and end of **requirements phase**
* Beginning and end of **design phase**
* Beginning and end of **coding**
* Beginning and end of **testing**
* **Before test plan approval**

✅ This ensures defects are detected **early** at each stage.

**Review Planning**

**📦 What should be included in a review package?**

1. **Document** to be reviewed
2. **Agenda** and **objectives**
3. **Facilitator's name**
4. **Roles** of all attendees
5. **Entry/Exit criteria**
6. **Review location, date, and time**
7. **Defect classification scheme**
   * Type (syntax, logic, performance)
   * Severity (critical, major, minor)
   * Origin (design, code, test plan)
8. **Procedures** for raising and handling issues
9. **Issue-handling procedures**

**Review Meeting Process**

**Structure of the Meeting:**

* **Facilitator** introduces agenda and participants
* **Author** presents the document
* **Reviewers** ask questions, raise issues
* **Recorder** logs defects with severity/origin
* Assigns issues for resolution with deadlines
* Recommendations are documented

**🟢 During the Meeting:**

* **Facilitator** introduces the review
* **Author** presents the document
* **Reviewers** raise issues based on prior review
* **Recorder** logs every issue/defect
* Assign responsibilities for each item raised
* Set **deadlines** for rework and resolution

**Decisions at End of Review**

After discussing issues, attendees decide to:

| **Option** | **Description** |
| --- | --- |
| ✅ Accept | Product is acceptable as-is |
| ❌ Reject | Product has critical flaws and must be redone |
| ⚠️ Provisionally Accept | Product has minor issues; can proceed with rework |
| 🔁 Schedule Follow-up Review | Another meeting needed to verify fixes |

**Rework & Follow-Up**

**🔁 Rework:**

* **Author fixes** all defects
* Ensures updated version is shared

**📋 Follow-Up:**

* Confirm all **discrepancies resolved**
* Ensure **exit criteria** are met
* Record **lessons learned**
* Document final report

**Mini Project / Practical Exercise**

Your slides recommend a **mock review exercise**. Here's a summary of how to carry it out:

**🔧 Setup:**

1. Select a **real software application** (e.g., a mobile app, website)
2. Assign **review roles** among team members

**🧐 Analyze:**

* Look at **UI/UX**, **navigation**, **functionality**, **performance**, etc.
* Take **notes and screenshots**

**📝 Report:**

* Prepare a **brief report** identifying:
  + Strengths of the software
  + Weaknesses or usability issues
  + Suggestions for improvement

Let me know when you're ready to continue to the next topic in your syllabus (e.g., **SQA Techniques** or **Project Planning & Estimation**). I’ll continue in the same detailed, structured way.