



QUALITY STATEMENT

MINITAB'S COMMITMENT TO PRODUCT & SERVICE QUALITY

REVISION DATE: JULY 2023

CONTENTS

Company	4
Quality Policy.....	4
Mission and Guiding Principles	5
Software Development Process	5
Requirements Building	6
Scoping and Discovery.....	7
Development and Test	7
Product Delivery, Maintenance, and Support	7
Software Quality Management	7
Quality Requirements	7
Software Verification and Validation	8
Functional Testing	8
Non-Functional Testing	9
Test Automation.....	10
Numerical Accuracy.....	10
Accuracy of Simulation Functionality	11
Targeted areas for calculation testing in simulation	11
Automation Strategy	12
Build and Code Quality Protocol	12
Design Reviews.....	12
Software Prototyping	12
Coding Guidelines.....	12
Code Reviews	12
Continuous Integration	12
Static and Dynamic Code Analysis Tools	13
Quality Gates	13
Process Control and Improvement.....	13
Configuration Management	13
Source Control.....	13
Version Control	13
Multiple Configurations	13
Security.....	13
Document Control	13

Security, Privacy, and Compliance	14
---	----



To Our Valued Customers:

At Minitab, we recognize that our high-quality software products must meet the needs and expectations of our customers. While Minitab produces commercial off-the-shelf software and provides Software as a Service solutions, we recognize that many of our customers desire information on how we develop our software and supporting services. To that end, we have created this document outlining the processes and practices we use to ensure the accuracy and reliability of our software. Included is a description of our Quality Policy, Mission, and other business processes.

If you have any additional questions regarding our policies and processes, please contact our Technical Support by visiting <https://www.minitab.com/en-us/support/>.

Thank you for using Minitab software!

COMPANY

In 1972, three Penn State professors created Minitab Statistical Software to help students perform statistical calculations easier, so they could concentrate more on statistical applications. Over time, the software was rapidly adopted by universities and high schools and across industries and became the market leader for analytics used in quality and continuous improvement fields.

Fast forward to today, and Minitab is still revolutionizing the analytics market. Minitab solutions help companies with everything from large scale digital transformation initiatives to focused projects on improving supply chain or eliminating costs from a specific area. Minitab has a broad range of solutions aimed to help organizations solve problems better, faster and easier than before.

Minitab has a very loyal customer base, and our customer list features over 90% of the Fortune 100 and the majority of the Fortune 500. Companies that use our products and services span across industries, sizes, and geographies. The one thing they share is their commitment to excellence and our goal is to help them achieve it!

QUALITY POLICY

For more than 50 years, Minitab has helped people discover valuable insights in their data by delivering exceptional software and unparalleled services. We make data analysis and problem solving easier. To remain on the cutting edge of statistical software and maintain the customer relationships we value so much, our products are constantly evolving and improving.

Our research and development efforts encompass not only statistical advancements and software innovation but also quality initiatives and customer service. Through decades of progress and refinement, one thing remains the same: our unyielding commitment to you, our customer.

As part of our commitment to you, Minitab aims to deliver the most reliable high-quality products, backed by the industry's best customer service. We invest heavily in the quality and dependability of our products because we know your data is important.

We recognize that the software products we create must meet the needs and expectations of our customers within the constraints of feasibility, integrity of the product, and viability of the company. They must be the right products, and they must operate in the right way.

We also recognize that the quality of software is multidimensional. Quality is not limited by one definition. In creating software, we seek to balance all aspects of high quality. We create software products that:

- Are available to customers in a timely fashion
- Are reliable and accurate in their operation
- Present an intuitive interface for both input and output
- Contain features that help customers solve problems and discover valuable insights in data

Finally, we recognize that the pursuit of quality is a journey with no end. We will continuously improve both the processes used to create our software products and the software products we create for our customers.

MISSION AND GUIDING PRINCIPLES

Our mission is to help people discover valuable insights in their data by delivering exceptional software and unparalleled services. Our guiding principles include:

EXCELLENCE

Our products and services will surpass expectations of quality and performance and will clearly distinguish Minitab from others in our markets.

ACCOUNTABILITY

We will encourage and inspire one another and hold ourselves and each other accountable for meeting schedule and quality expectations in support of our fundamental objective.

TEAMWORK

We will collaborate, cooperate, and work across the organization to create better products and deliver solutions more effectively.

AUTONOMY

We will give each other the resources, trust, and flexibility we need to complete projects and tasks successfully, and opportunities to contribute to our fundamental objective in meaningful ways.

COMMUNICATION

We will encourage the open exchange of ideas and perspectives and will share information to help each other understand how our work contributes to our fundamental objective.

BALANCE

We will encourage each other to maintain a healthy balance in our lives and be supported with company policies that provide the opportunity to fulfill both our professional and personal obligations.

INTEGRITY

We will treat everyone we encounter in our work with honesty, respect, and compassion.

SOFTWARE DEVELOPMENT PROCESS

Minitab uses an Agile Software Development Life Cycle that is focused on iterative software development and is driven by feedback and market research, all in support of our organization's mission and vision.

Reviews occur throughout development, both by customers and by customer-facing representatives to ensure that expectations and requirements are met. In addition, there is traceability throughout our development process, from our mission and vision to our product plan, through requirements, design, and development.

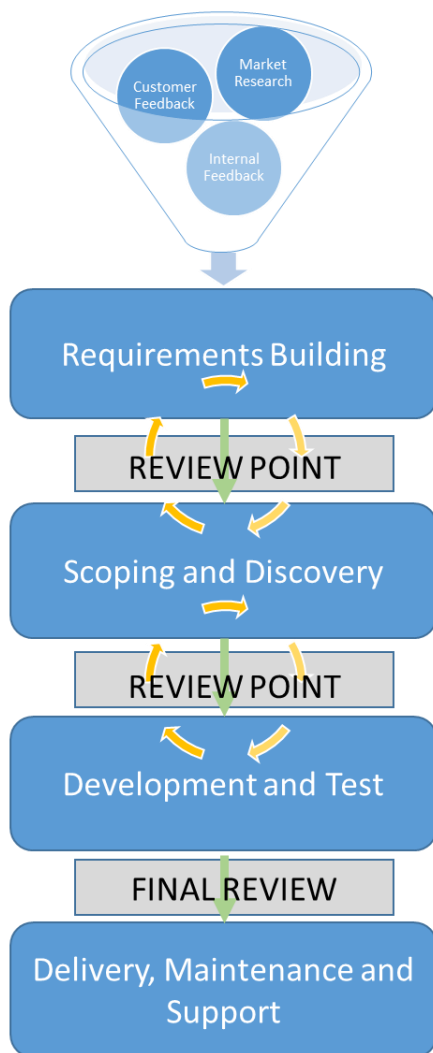


FIGURE 1: High-Level Software Development Process

The sections below provide more detail on each of the main areas of software development.

REQUIREMENTS BUILDING

During requirements building, using customer feedback, market research and internal feedback, a repeatable process is followed to outline product and release requirements. This information is captured in both the product plan and supporting requirements documents. Each release has an associated project charter or release plan which documents the deliverable and requirements for the software.

These requirements may include:

- Prioritized feature lists
- Quality requirements
- Delivery requirements
- Budget requirements
- Regulatory and/or compliance requirements

- Technical requirements
- Cyber security requirements

Once finalized, Product Management works with the Program Development Manager to source the development team and begin scoping and discovery.

SCOPING AND DISCOVERY

During scoping and discovery, lead engineers evaluate the requirements and outline the feasibility, resource requirements and any technical risks that may exist, along with mitigation strategies. Initial designs, development strategies, and project plans are compiled. Using this information, the organization decides whether to pursue the development of the project.

DEVELOPMENT AND TEST

During development and testing, the deliverables for the project are fully designed, developed, and tested. This cycle is iterative, and each of these aspects is continually refined during development. Testing and validation occur throughout development, including the use of automated continuous integration systems to ensure successful integration and stability of our codebase. All project requirements and design are verified prior to final release.

PRODUCT DELIVERY, MAINTENANCE, AND SUPPORT

During the software release phase, final testing occurs (including Installation and Deployment testing), and the release is assessed for readiness. In addition, internal training is provided to ensure that customer-facing groups are prepared to support the release.

Once released, the software is supported in the field until its end of support date. As issues and enhancements are identified, they are documented and evaluated for inclusion in a future release of the software. Each issue is logged in a tracking system and tracked through resolution.

For more information on Minitab's technical and documentation support, including our support policy, please see:

<https://www.minitab.com/en-us/support/>

SOFTWARE QUALITY MANAGEMENT

Minitab's unyielding commitment to customers includes our pledge of quality and accuracy. This commitment to software quality is a shared commitment and is the responsibility of the entire organization. Each team takes ownership of the result and the requirements for quality form the foundation of the software development process and are tied directly to each development activity. The following sections detail our software quality requirements, processes, and methodologies used to support quality management at Minitab.

QUALITY REQUIREMENTS

Minitab utilizes the ISO/IEC 25010 framework to define product quality requirements. These requirements are based on the following characteristics:

- Functional Suitability
- Performance Efficiency
- Compatibility

- Usability
- Reliability
- Security
- Maintainability
- Portability

SOFTWARE VERIFICATION AND VALIDATION

Software verification and validation are crucial aspects of the development process at Minitab and are the responsibility of the entire development team. Verification and validation occur throughout the development process and conclude with a final testing phase to ensure compliance prior to delivery.

A combination of the following methodologies is utilized in the testing of Minitab applications.

FUNCTIONAL TESTING

Functional testing is performed to verify the functional requirements of the application. The following methods are used, as specified, within Minitab development:

VALIDATION OF STATISTICAL AND GRAPHICAL RESULTS

All statistical and graphical output is validated for market readiness using one or more of the following methods. A requirements traceability matrix outlines Minitab Statistics Software commands and their associated automated test file directories used to validate results against requirements and design. The matrix is a part of the validation kit, which is available by request.

- Validated by hand or computed calculation
- Validated by comparison to other software
- Validated by comparison to textbook examples

WHITE-BOX TESTING

White-box testing strategies focus on the internal structure of an application, focusing on individual classes or subsystems. The following methods are used:

- Unit Testing
- Integration Testing
- Code Coverage

BLACK-BOX TESTING

Black-box testing focuses on the functionality of the application, without knowledge of the internal structure. It typically focuses on requirements and customer use cases to define test cases. This includes the following types of testing:

EXPLORATORY TESTING

During exploratory testing, the tester works with the application and continually develops test cases based on the outcome of the testing. Defects found during exploratory testing are typically added to a formal test plan or automated test.

SCENARIO TESTING

Utilizing user scenarios and data, the tester validates that the software performs correctly under user conditions. This often involves testing a feature or set of features including the calculation, relevant dialogs, and outputs.

SYSTEM TESTING

During system testing the complete, integrated system is tested against specified requirements.

REGRESSION TESTING

During regression testing, both automated and manual testing are performed against the software to ensure that no new errors were introduced into existing functionality. Regression testing is done at specified points during development and during final test. It typically includes the complete automated testing suite along with a series of manual tests.

ALPHA TESTING

During Alpha testing, a subset of the application is evaluated by internal or external users. Feedback is then provided to the design and development teams and necessary changes are incorporated into the release. There can be multiple Alpha tests performed during development.

BETA TESTING

During Beta testing, external customers are asked to evaluate the release, which can include documentation and the installation process. Feedback is evaluated and incorporated into the current or future releases, based on priority.

NON-FUNCTIONAL TESTING

Non-functional testing encompasses all validation for external factors impacting the application. It validates that the software will work consistently and predictably in the field.

PERFORMANCE TESTING

The performance of the application is evaluated and monitored under varying conditions. This includes performance of:

- Algorithms
- Rendering of outputs
- Key aspects of the user Interface

LOAD TESTING

Load testing is performed to ensure that cloud-based systems perform under expected user conditions (or load). Using a specified combination of system and load configurations, the following performance and capacity indicators are assessed:

- Average response time
- Error rate
- Time elapsed between view updates
- Average CPU usage
- Memory leaks
- Peak load
- Concurrent user testing

PENETRATION TESTING

Minitab maintains both internal and external penetration testing programs. For more information about our penetration testing programs, please see our Minitab Product Security Protocol document.

COMPATIBILITY TESTING

Testing of the complete system is performed against defined configurations, which include:

- Supported operating systems and browsers
- Hardware configurations
- Supported third-party products
- Ancillary applications
- Visual settings
- Compatibility between cloud and desktop versions

INTERNATIONALIZATION AND LOCALIZATION TESTING

During development the architecture is evaluated for ease of translation. Once translated, the software is validated using:

- Regression tests run on selected platforms, languages, and operating systems
- Automated and/or manual tests run on select features of the application

INSTALLATION AND DEPLOYMENT TESTING

Testing is performed to ensure that the programs, files, and processes are in place to make the application available throughout its lifecycle. This includes the testing of installation, deployment, licensing, software updating, interoperation, and uninstallation of our desktop applications.

TEST AUTOMATION

Each release is validated using a combination of manual and automated testing, depending on specific release requirements. The automated suite is systematically run to regression test all new versions and conversions of the software.

Automated tests may be used to validate the following aspects of the application:

- Verification of statistical results
- Verification of individual classes or subsystems within the application
- Verification of GUI elements
- Scenario tests that validate paths through the application
- Verification of command and application performance

NUMERICAL ACCURACY

To reach our goal of making data analysis easier, we must be able to deliver accurate and reliable results. One of the most important quality issues in the development of analytical software is numerical accuracy. Accuracy is the degree of closeness between a reported result and the true value.

The nature of hardware and software presents a set of limitations that can affect the accuracy of a computed value. One limitation is due to hardware architecture and the finite memory of computer systems. It is necessary for computers to perform their calculations using numbers of finite precision; however, not all real numbers can be represented in binary

finite precision. Due to this limitation, the results of certain arithmetic operations may include numerical variations caused by truncation.

Another factor impacting accuracy is the selection of algorithms and the techniques used to implement those algorithms in the software. Designers and developers must carefully choose and implement stable algorithms that prevent the amplification of any rounding errors brought upon by the finite precision limitation. Accuracy and efficiency must both be factored into the algorithm selection and implementation processes to achieve optimal performance.

The determination of a value's accuracy typically requires comparison with a known standard.

To that end, Minitab utilizes multiple methods to validate the accuracy of statistical and graphical results. These include:

- By hand or computed calculation
- By comparison to other software
- By comparison to textbook examples

ACCURACY OF SIMULATION FUNCTIONALITY

Simulation functionality in Minitab Engage and Minitab Workspace is tested from both the user interface as well as from code in the form of automated test cases geared toward the verification of computational functionality. A complete traceability matrix of the Monte Carlo simulation test files with their target testing areas is available by request.

TARGETED AREAS FOR CALCULATION TESTING IN SIMULATION

SIMULATIONS

These test cases verify random number generation is being done correctly and consistently, summary statistics are calculated correctly, values used in sensitivity analysis are calculated correctly, formula operators are interpreted correctly, and formula functions are calculated correctly.

OPTIMIZATION

These test cases verify the stochastic search methods we use are executed correctly and consistently and that our heuristic approach is correctly identifying global optimal values over the search range. There are a few test cases that have been validated as incorrect, and this is a natural result of some stochastic search methods being incapable of identifying optimal conditions for some highly complex or intentionally deceptive test landscapes.

DISTRIBUTION FITTING

These test cases verify that data sets used in distribution fitting produce the correct results for both the distributions that are recommended to the user as well as the parameters of those distributions.

PERFORMANCE

These tests set timing benchmarks for data sets of various sizes in simulation, optimization, and distribution fitting.

AUTOMATION STRATEGY

Test cases are automated and run anytime code is changed in the simulation repository. Values in the auto-generated output files are compared using custom comparison code that parses output JSON and compares numeric values for equality to a precision of 1e-12.

Each time a new formula function, optimization objective or strategy, summary statistic, or distribution type is added, new test cases are generated and verified to ensure validated results do not change unexpectedly for all value-added simulation functionality.

BUILD AND CODE QUALITY PROTOCOL

Product quality at Minitab reaches far beyond the testing function. The Software Engineering staff takes total ownership of their code quality and employs many methodologies and protocols to ensure a successful delivery.

DESIGN REVIEWS

Design reviews are an examination of the design and the requirements for a specific project. Before the coding begins, the developer assigned to the project reviews the design, identifies risk areas, and establishes timing and sizing budgets. Action items are identified, assigned, and, if necessary, resolved before coding begins.

SOFTWARE PROTOTYPING

Risk areas identified during a design review may be prototyped before or in conjunction with project implementation. Examples of risk areas include new algorithms, large datasets, and new technologies.

CODING GUIDELINES

Industry standard coding and framework guidelines are used as a basis for coding guidelines. These guidelines enable developers to create more uniform and maintainable code. Adherence to the coding guidelines is enforced through code reviews and code analysis tools. Major areas that are addressed in the guidelines include naming conventions, file organization, resource and settings file usage, and overall style.

CODE REVIEWS

Code reviews are a formal examination of source code intended to identify and resolve coding errors, security vulnerabilities, inefficient code, unused code, or syntax in violation of the coding standards. The lead developer determines the extent to which code reviews should be completed during development and the process is handled electronically using pull requests within our source control management system. In addition to formal code reviews, less formal review of the code is ongoing through pair programming and information exchange between the developers.

CONTINUOUS INTEGRATION

During development, a continuous integration system runs automated tests against each change to the software source code. These tests include unit, integration, and algorithm checks that compare calculated results to archived results to ensure that the mathematical results of the software have not changed. This ensures that changes made do not negatively impact the code quality.

In addition, tools are used during a pull request to analyze and identify when programming standards are not met. Violations are treated as errors, and the code cannot be added to the release baseline until the defects are resolved.

STATIC AND DYNAMIC CODE ANALYSIS TOOLS

Various third-party tools are used to analyze code, identify potential errors, security vulnerabilities, or coding standards violations, and achieve a consistent development style across projects. For more information about static and dynamic code analysis, please see our Minitab Product Security Protocol document.

QUALITY GATES

Quality gates are used to establish minimum acceptable levels of quality for all functional and non-functional requirements of a release, including algorithmic results, usability, and security. The project team, along with stakeholders, assess quality levels throughout development and prior to release. Some requirements, such as security, require a heightened level of compliance and review prior to product release.

PROCESS CONTROL AND IMPROVEMENT

Various tools are in place within development to ensure adherence to a consistent and effective development process. These include:

- Detailed development checklists
- Detailed testing checklists
- Peer reviews of code, tests, and documentation

Process rigor and continuous process improvement are integral to development activities at Minitab. Issues found during development and post-release are analyzed for trends. Root cause analysis is then performed to identify potential process improvements. Product quality, along with continuous process improvement activities, are submitted to senior management monthly for review.

CONFIGURATION MANAGEMENT

Minitab utilizes configuration management to ensure accurate tracking of changes and to control the baseline for both source code and supporting documentation.

SOURCE CONTROL

VERSION CONTROL

The configuration-management system maintains version control which includes date, responsible individual, and a comment field. A history of changes to each file in source control is maintained.

MULTIPLE CONFIGURATIONS

Source code is maintained for each version of a release via source control branching. If multiple versions of the application are developed simultaneously, branches exist for each individual configuration.

SECURITY

Changes to source code and automated test scripts are limited to trained personnel. Unauthorized users are restricted from submitting file updates.

DOCUMENT CONTROL

In addition to source code, necessary project documents are controlled using the following guidelines:

- Each project document is maintained electronically on the internal project site and is available for review by all team members
- The current version of a document is the only valid version that is used within product development

SECURITY, PRIVACY, AND COMPLIANCE

At Minitab, we are committed to building trust by making your experience with us safe and secure. We recognize that having all the essential security, privacy, and compliance strategies in place is critical in protecting you and the solutions we share. Please reference the [Minitab Trust Center](#) for the latest information on privacy, compliance and security for our products and services.