

# Custom Software Development



## Our Approach

MAT executes our approach to Software Development engineering in the following phases:

- ⇒ Requirements Gathering
- ⇒ Design
- ⇒ Development
- ⇒ Qualification Testing
- ⇒ Integration

### Requirements Gathering

To ensure software quality and completeness, MAT first creates a software requirements specification (SRS) — identifying requirements and business rules for fixes, modifications, and enhancements. For version control, we store the SRS in TortoiseSVN, a real-time Software Configuration Management (SCM) tool.

MAT next creates a requirements traceability matrix to cross reference requirements and maintenance/enhancements with Help Desk (HD) Change Items (CI) and other associated work processes such as design, validation, unit testing, and more. Our software development team also leverages project management and bug-tracking tools to help allocate work items to resources, provide cost containment, improve efficiency, and build an overall superior application. During requirements gathering, MAT maintains continuous communications with the stakeholders to ensure the requirements meet quality standards throughout the software development process.

### Design

During software design, MAT collaborates with customers to incorporate end user behaviors into the design of the product. Our collaborative approach allows our team to react directly to functional, technical, and design challenges that may emerge. MAT places a very high value on product and process quality. For this reason, all designs go through an extensive review process to ensure that the design will satisfy our customer's needs.

### Development

MAT implements the design using industry proven techniques. Each module of system design is converted to source code and tested separately at the unit and module levels. As modules/components are completed, they are moved into an integration/system test phase to further validate that the module/component will function properly when integrated into the whole product. During this process, MAT collaborates with stakeholders to produce and test incremental "builds", as opposed to creating an entire solution at once. Product demos are provided during this process to ensure software usability and an approved function, look, and feel, as well as to ensure that it product performs as the customer desires. During development, MAT ensures and predicts progress and schedules using:

- A single coding standard
- Test-Driven Development (a proven technique that ensures that tests assets are designed according to specifications before the actual unit of the product is built, so that the unit can be tested for conformance to the specifications)
- Secure and continuous integration with a repository driven by Subversion version control. This system allows the development team to instantly become aware of any violations and/or defects introduced by coding changes as they occur, so that they can be corrected before proceeding
- Adherence to structural design models for needed business rules and constraints
- Small iterative development cycles to allow for adaptation of evolving requirements
- Daily stand-up meeting to ensure accountability and to manage risks as they occur
- Peer code reviews and collaboration to inspect usability and compliance with established standards

### Testing

Testing for software fixes or added software functionality is performed in a three-phased approach:

- Unit Testing - Tests only the unit code. Passed test cases will be committed along with the source code to the source code repository.
- Module Testing - Tests the changed code's affected vertical and horizontal areas, ensuring that the modified code will not negatively impact other related areas.
- System Testing - Tests the system-wide quality of software development fixes/changes to ensure that they perform according to specifications.

### Integration

MAT documents and provides to the level required by the customer, appropriate documentation and resource materials to accompany the software applications. MAT can also provide certified personnel to assist in integration of the software into the customer's environment, as well as Training Support Packages (TSPs), depending on the actual operational environment.

## SAMS-E Project

MAT has the proven ability to provide custom platform, application, operating system, and system support software.

In April 2004, MAT won its first major software development project, a government contract to modernize three Army logistics systems — the Unit Level Logistics System-Ground (ULLS-G) and Standard Army Maintenance System-1 & 2 (SAMS-1& 2) — into the Standard Army Maintenance System-Enhanced (SAMS-E).

To accomplish this task, MAT:

- Converted multiple outdated systems into one modernized system
- Successfully performed data migration
- Continued functionality between outdated systems and the modernized system
- Maintained the platform
- Ensured successful data exchange
- Provided support and functionality to the new SAMS-E system

MAT meticulously studied legacy data, carefully mapped data fields from obsolete systems to the modernized system, and precisely validated data to new standards by writing rules and ensuring smooth data migration. MAT provided database structure design for combined databases, technical support for database platform conversion, and modifications to applications as required for the conversion.

MAT currently fields and maintains SAMS-E software to Army units worldwide and provides ongoing training and technical assistance that supports Army maintenance and logistics systems in daily operations. Fielding SAMS-E to the Army was a project extensive in size, scope, and complexity—encompassing over 14,000 systems worldwide in all levels of Army maintenance.

## Our Qualifications

In February 2008, MAT has achieved the Software Engineering Institute's (SEI) Capability Maturity Model® Integration for Development (CMMI-DEV) Level 3 Appraisal rating. We incorporate the CMMI model into our software development methodology as a measure of quality control, quality assurance, and risk management. Our team incorporates best practices identified in Institute of Electrical and Electronic Engineers (IEEE) 12207 framework for acquiring and developing software. Each of these engineering disciplines as applied influences the other by seeking to expand the scope of each discipline's reach. These achievements demonstrate MAT's continued commitment to improved software engineering and high-quality customized frameworks.

### Programming Skill Set

MAT's current software programming languages and certifications include, but are not limited to, the following:

- C#
- Java
- PHP
- Ruby
- Oracle
- PL/SQL
- MS SQL Server
- T-SQL
- Project Management Professional (PMP)
- Microsoft Certified Application Developer (MCAD)
- Oracle Certified Professional (OCP)
- Microsoft® .Net Framework™ v1.0, 1.1, 2.0, 3.5, 4.0
- ASP.Net™ web application development framework
- PHP web development framework
- Linux software development
- Web Sphere™ messaging middleware
- AMQP universal messaging framework
- Windows Communications Foundations™ consolidated enterprise application development toolkit
- Windows Presentation Foundation™ user interface development framework
- Windows Forms™ user interface development framework
- Microsoft Silverlight™
- And more