mendix

Presented by Faezeh Khorram

Contents

- Introduction
- Application Development Lifecycle
  - Ideate
  - Develop
  - Deploy
  - Operate
- Quality Assurance
  - Feedback Mechanism
  - ATS: Application Test Suite
  - AQM: Application Quality Monitor
  - APM: Application Performance Monitor
- Model consistency
- Reusability mechanisms
- Overview

Introduction

No-code and Low-code in one

Collaborative Visual Development

Multi -Channel Apps: Build applications once and run them anywhere

Cloud Native and Stateless Architecture: Scale vertically or horizontally

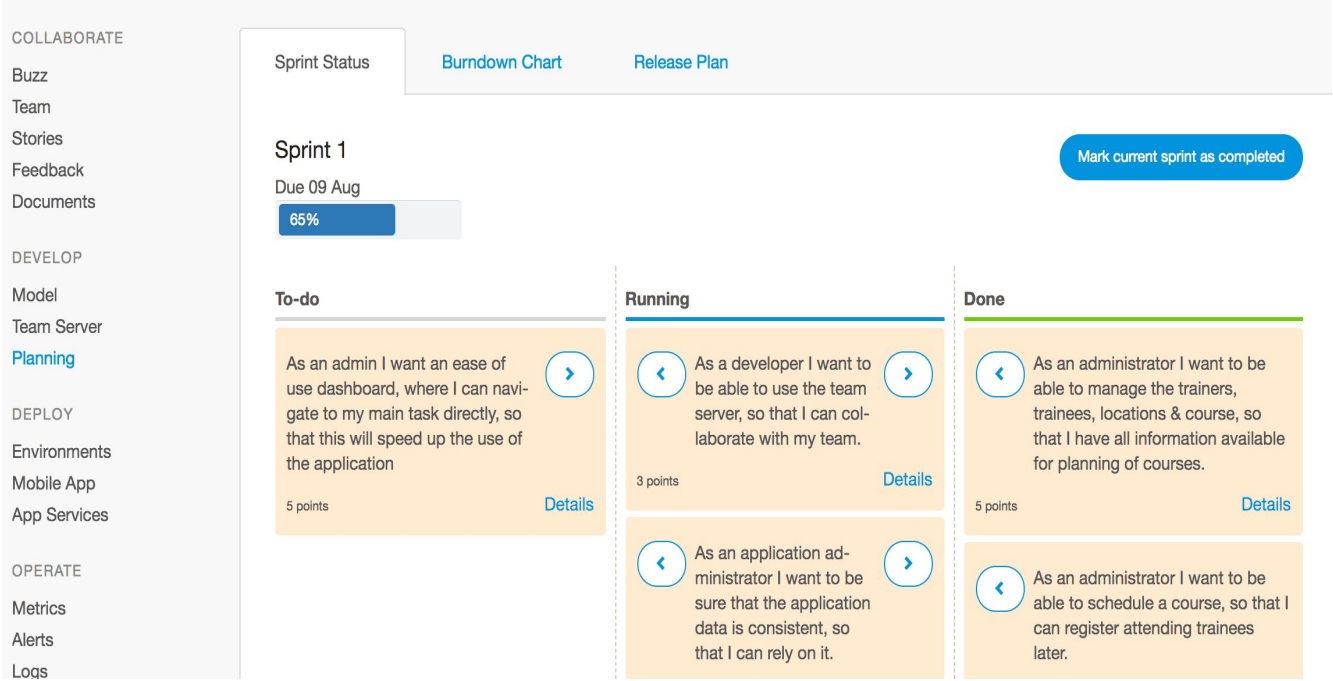
Open & Extensible (API, SDK)

Unmatched Time to Value: 10x faster development, 70% fewer resources

Application Development Lifecycle:
Ideate, Develop, Deploy, Operate

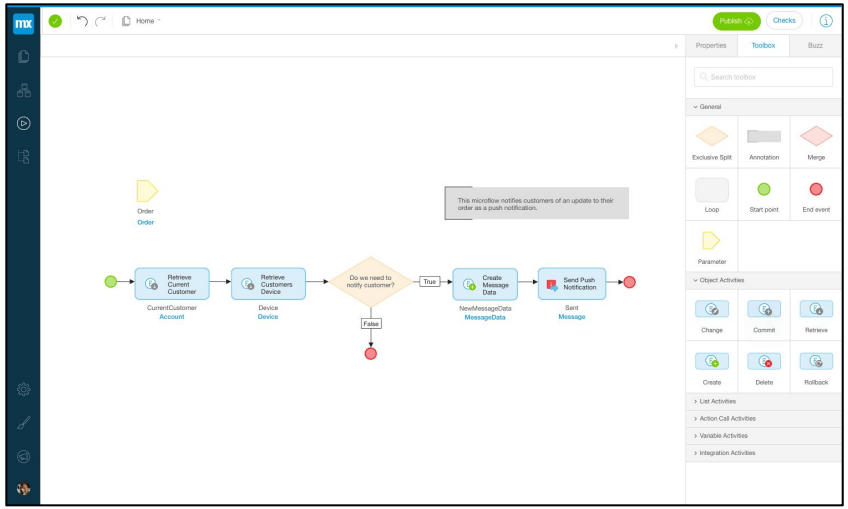
- Start with existing applications or templates, or from scratch
- Collaboration between stakeholders and developers from the first until the end
- Requirement elicitation using user stories
  - Definition of new stories
  - Pulling from existing project management tool by ‘stories API’
- Planning in terms of assembling Backlog and Sprint plan
- Requirement traceability through built-in feedback management

Ideation- Planning based on Scrum

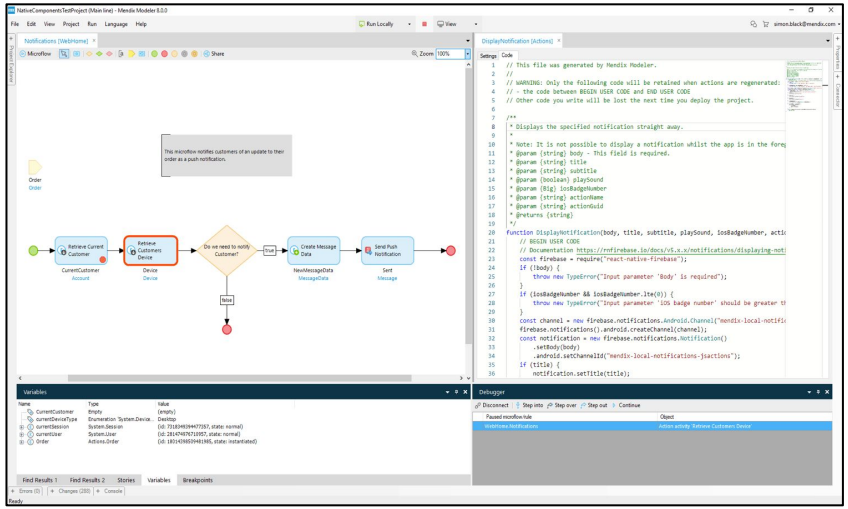


Application Development Lifecycle:
Ideate, **Develop**, Deploy, Operate

No-code: Drag & Drop web-based Mendix studio



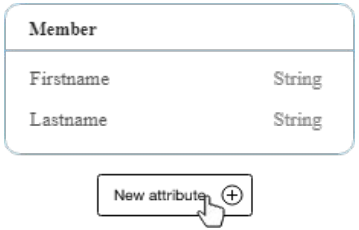
Low-code: Studio pro for experienced developers



Development- Visual Modeling

Domain Modeling

Build objects, attributes, validations, and data models with a UML-based data modeler

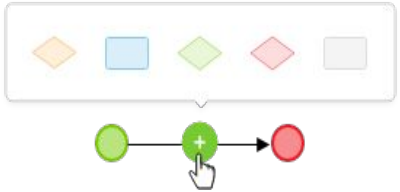


UI Modeling

Use a WYSIWYG graphical page designer to build fully-responsive web and mobile user interfaces

Workflow Modeling

Create client-side business logic, offline mobile apps, dynamic highly-responsive user interaction



Live Debugging

Isolate issues, visually step through logic and interrogate data to rapidly solve application bugs

Application Development Lifecycle:
Ideate, Develop, **Deploy, Operate**

Deployment:

- To a cloud of user`s choice:
  - Mendix public cloud (runs on AWS)
  - Private cloud (On-premises infrastructure)
- Support DevOps and CICD pipeline
- Integrate directly with the user`s toolchain via the platform APIs

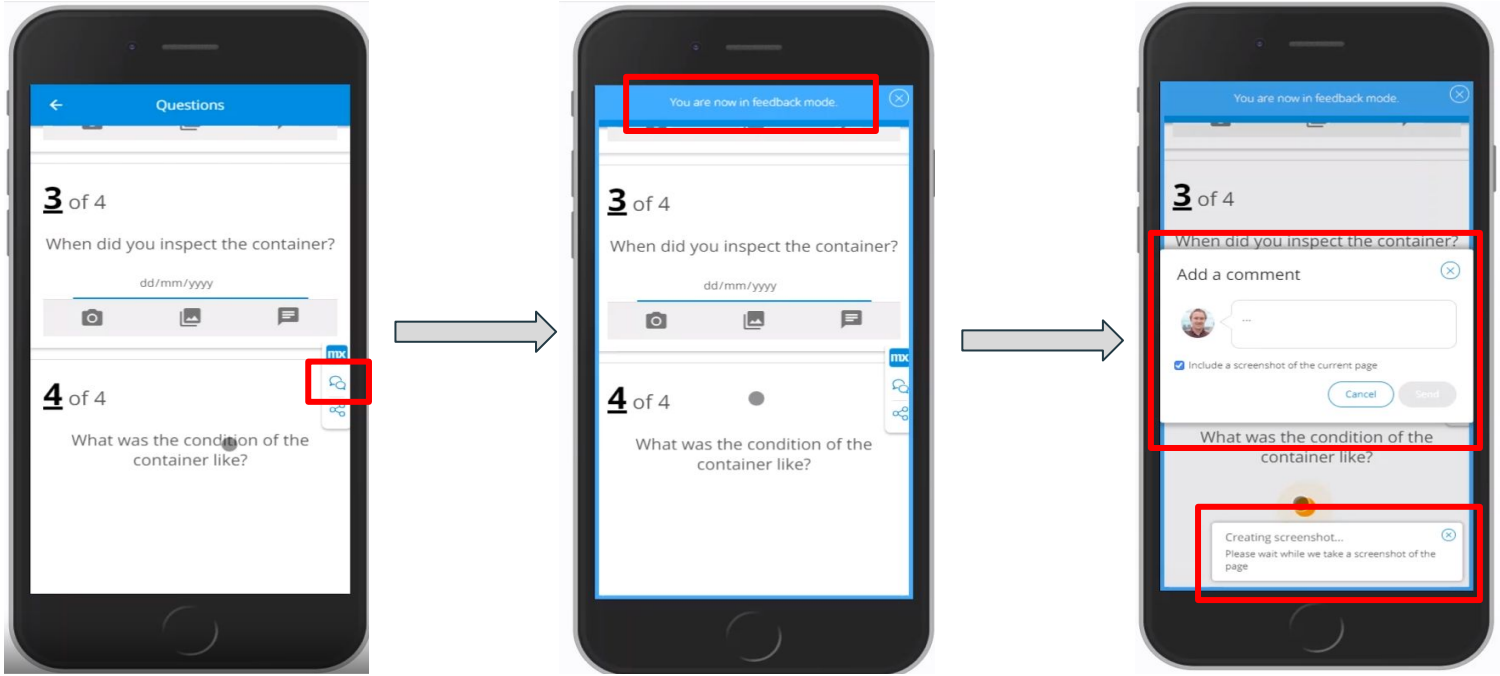
Operation:

- Monitor and manage the entire portfolio
  - Configure the environment
  - Monitor real-time data
  - Manage backups, usage and user access

Quality Assurance

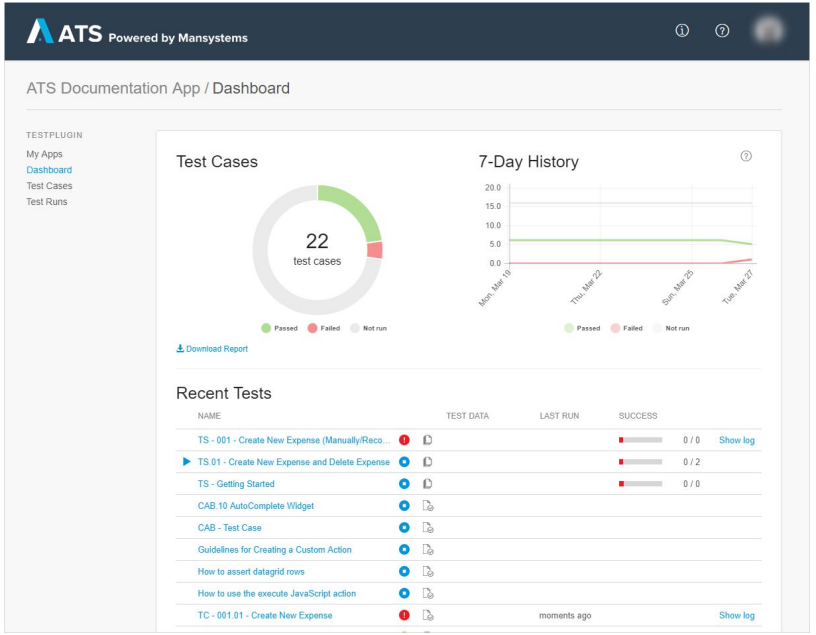
- Business and IT Alignment
- Embedded Testing: Testing throughout lifecycle
  - Functional Testing and Regression Testing -> Application Test Suite (ATS)
  - Unit Testing -> ‘UnitTesting’ module in the Mendix App store
  - Integration Testing -> SOAP UI
- Application Quality Monitor (AQM)
- Application Performance Monitor (APM)
- Real-time error checking
- Automated Consistency Checking
- AI-Assisted Development: Give suggestions to best next steps based on standards and by using milion of existing application models
- Third-party test tools and services: Support Selenium and Junit testing frameworks

Feedback Mechanism



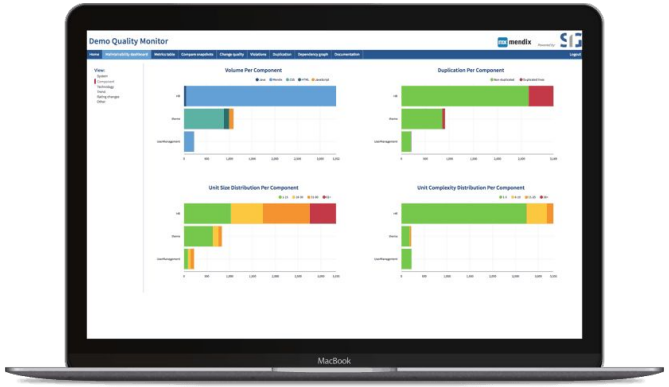
Application Test Suite (ATS)

- A suite of tools built by Mansystems on top of Selenium
- Test design
  - Test case templates
  - ATS Recorder: a chrome plugin
- Linking test suites/test cases to user stories
- Reporting: Error log, Screenshot, Records
- Scheduling: run test at any time without manual input
- Parallel test execution
- Bulk testing: upload data from CSV and Excel

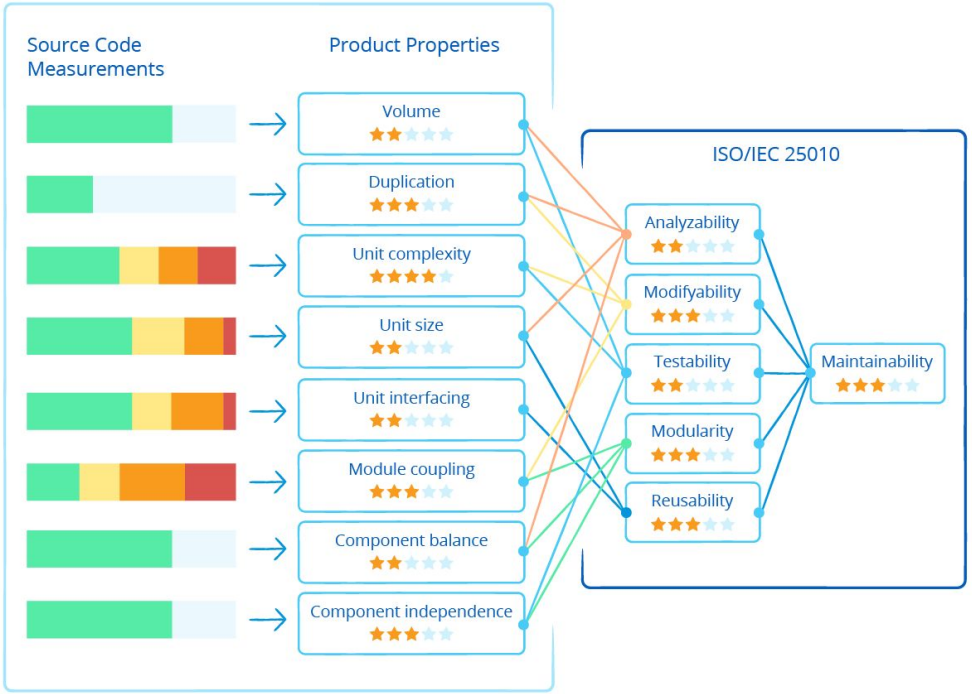


Application Quality Monitor (AQM)

- Static analysis of application models on a daily basis
- Measure maintainability based on ISO 25010 standard
  - Use of SIG mathematical engines
- Rating the application against a database of thousands of projects on a scale of 1-5
- Highlighting any potential issues that should be addressed



AQM-Application of ISO 25010 in Mendix



Application Performance Monitor (APM)

- **Trap Tool**: Continuously records all levels of logging and stores it when an error occurs
- **Statistics Tool**: Measures the durations of microflows and browser-client requests
  - Gaining insight into what a system is doing when it doesn't perform optimally
  - Determine potential performance issues by looking at the trends
- **Performance Tool**: Analysis of individual functions and visualize them where improvement is possible
- **Measurements Tool**: Monitoring CPU and memory, performing measuring queries, and altering when a limit is exceeded

APM

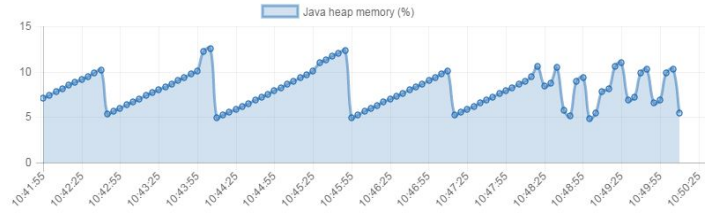
Dashboard

- Statistics Tool
- Performance Tool
- Trap Tool
- Measurements Tool
- JVM Browser
- Query Tool
- Log Tool

Statistics

Microflow name	Count	Avg (ms)	Max (ms)	Total (ms) ▼
TestCase.TaskSimulator_ScheduledE...	2	236	240	472
TestCase.GetRandomTask	9	26	35	237
TestCase.CompleteRandomTask	6	32	42	194
TestCase.ProgressRandomTask	3	32	37	98
TestCase.GetRandomUser	9	6	14	58

Measurements



Traps

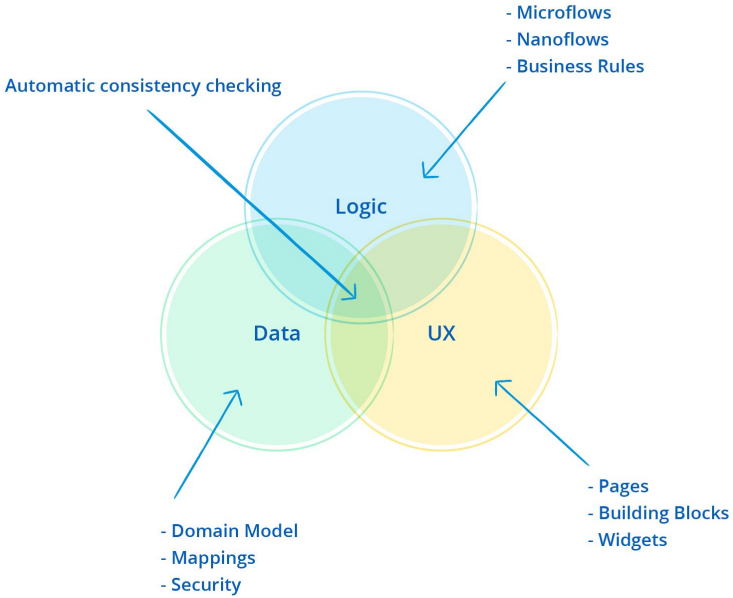
Last traps	Timestamp
Memory > 10% in Java heap memory (%) with value 10.9483655485861971...	2016/11/28 10:49
Memory > 10% in Java heap memory (%) with value 10.5528829648448228...	2016/11/28 10:48
Memory > 10% in Java heap memory (%) with value 10.0681634959006368...	2016/11/28 10:47
Memory > 10% in Java heap memory (%) with value 10.0292915107456742...	2016/11/28 10:45
Memory > 10% in Java heap memory (%) with value 10.0633419532430135...	2016/11/28 10:43

Logs

Log level	Timestamp ▼	Message
Info	2016/11/28 12:41	Started dynamic scheduled event Task Simulator
Info	2016/11/28 12:32	Finished cleanup
Info	2016/11/28 12:32	Cleanup manual query audit
Info	2016/11/28 12:32	Cleanup unlinked metrics
Info	2016/11/28 12:32	Cleanup events

Model consistency

- **Consistency checks**: Errors (must be solved), warnings (can be ignored), deprecations
- **Single Modeling Environment**: Checks the completeness and consistency of the entire model
- **Pattern Recognition**: Performs analysis to detect patterns, such as incomplete decision trees
- **Regression Analysis**: Checks the impact of changes on the model
- **Model Refactoring**: Automatically and consistently propagated throughout the whole model



Reusability

Mendix allows reuse across the entire platform

- **Model-Driven**: Componentize application models to ensure reuse
- **Reuse Components**: Leverage the Public and Private app store to reuse components across applications
- **Microservices**: Build applications to leverage microservices architecture

Overview

- **Domain**: Native Mobile, Web
- **OS**: iOS, Android
- **Modeling Language**: UML, BPMN
- **Programming Languages**: Java, Java Script, HTML5, CSS3
- **Frameworks**: Bootstrap, React, PhoneGap
- **Database servers**: SQL, Oracle, IBM DB2, MariaDB, MySQL
- **Strategic partners**: SAP, IBM

Thank you for your attention!