

CASE STUDY: THE UTILIZATION OF LOW-CODE DEVELOPMENT TECHNOLOGY TO SUPPORT RESEARCH DATA COLLECTION

Robert L. Totterdale, Florida Gulf Coast University, rtotterdale@fgcu.edu

ABSTRACT

Research data must be collected and maintained in compliance with a myriad of laws and regulations that protect the privacy of participant's information, and should be captured in a manner that is cost effective and timely. This paper discusses research data collection, explores challenges and approaches for collecting data, and describes how low-code development technology can be utilized to facilitate the secure and efficient collection of research data in the healthcare industry. This paper is based on research being conducted in the healthcare industry but has broad applicability across other industries and research areas that collect personally identifiable information, or other sensitive data protected by U.S. or international laws and regulations.

Keywords: Low-code Development, Data Collection, Privacy, Research, Healthcare

INTRODUCTION

Because of changing laws and regulations both domestically and internationally, researchers must carefully identify the data they will collect and plan for how they will capture, process, analyze, share, and ultimately destroy the data over time. The management of data throughout the research process has been referred to as the research data lifecycle.

In the following sections, this report will provide a definition of data, will discuss how data relates to the IRB process and how laws and regulations affects the data lifecycle, and will identify challenges relating to research data processing. In addition to the discussion of data and its impact on research, this paper discusses low-code development and how it can provide a solution for researchers to expedite their data collection process in a secure way. An actual case example of the use of a low-code development platform is discussed to provide additional insight on the value of the technology in research.

Research Data Collection

Data used in research can be in many forms such as characters entered through a keyboard, biospecimens, images, video recordings, paper notebooks, and drawings. It can be captured through observation, machine generated algorithms, historical record abstracts, recordings, and through other physical means. At it's most basic level, data can be defined as "facts and statistics collected together for reference and analysis" (Data, 2018) and from a research perspective can be defined as that which "is collected, observed, or created, for purposes of analysis and to produce research results" (University of Edinburgh, 2018). Each stage of the research data lifecycle (Figure 1) has implications for the researcher to insure that adequate safeguards are in place for the protection of the data and is in compliance with local, state, federal laws, and organizational requirements.



Figure 1. Research Data Lifecycle (U.K. Data Archive)

In higher education, Institutional Review Boards (IRBs) must approve research involving human subjects. The IRBs assess protocols, consent forms, data collection forms, and various other aspects of the data lifecycle. Since 1991, review boards have operated under guidance from “the Common Rule- Federal Policy for the Protection of Human Subjects”. However, a revised Common Rule published in the Federal Register on January 19, 2017 has now been approved and applies to all research initiated after Jul 19, 2018 (Federal Register, 2017).

When preparing an IRB application for approval, researchers must address a number of questions that relate to data including how and where it will be collected, transmitted, and stored (Table 1). In addition, during planning, the researcher must consider the type of data being collected (sensitive, personally identifiable), how and when it will be destroyed and whether the data is available from public records, is anonymous, or is confidential (FGCU, 2017). As part of the IRB process, approaches for protecting the data such as encryption, the use of secure transmission protocols, and implementing authentication measures may be proposed as part of the protocol relating to electronic data capture, processing, and storage.

Table 1. IRB Questions

Data Lifecycle Component	Questions
Creating (collecting data)	who, what, where, how, type
Processing data	what, how (transformed)
Analyzing data	protocol, interpretation of results
Preserving data	what, where, how, types
Access to/destroying data	who, what, where, how

Legal and Regulatory Discussion

In addition to compliance with IRB requirements, federal, state, and local regulatory authorities have created a number of privacy laws that specify substantive obligations for those who work with or store personally identifiable, sensitive, or confidential information. For those found not in compliance with the laws, fines of over \$250,000 and or 10 years of imprisonment can occur. Although laws and regulations may vary by industry, geography and type of data, organizations and individuals must have awareness of the laws and adopt sound practices to protect the data they have collected. These laws also apply to researchers who work with the data. Examples of some key laws relating to data privacy are identified below in Table 2 (Whitman, 2017).

Table 2. Laws and Regulations

Law	Industry	Addresses
HIPAA	Healthcare	Standards for information disclosure and transmission
HITECH Omnibus	Healthcare	Increased protection of personal health information
Privacy Act	Government	Regulates government collection, storage , use and dissemination of individual personal information
Gramm Leach Bliley Act	Banking	Privacy and information protection
Sarbanes Oxley	Various	Accountability for executives in publicly traded companies
Computer Fraud and Abuse Act	Various	Accessing a computer without authorization
GDPR	Various	European regulation for data protection

Researcher Challenges

As research protocols are planned, a number of decisions must be made about data and the approaches that should be selected to satisfy the needs of the research while at the same time minimizing risk to inadvertent exposure. In addition, researchers must also consider the legal implications and risk as noted above, for not complying with various laws and regulations. An interrogatory approach can be used to help decide appropriate research designs by addressing the questions outlined in the following table.

Table 3. Design Approaches

Question	Design Approach
Who	Authenticate users for access to data and limit access to specific data based on role
What	Collect only necessary data. Make anonymous where possible
Why	Determine minimum data to meet objectives. Remove personal identifiers and de-identify by removing key data.
When	Destroy data at earliest date
Where	Determine storage locations (cloud/local host/ file cabinet). Determine where application will run
How	Provide data collection at the source with input validation Authenticate users by role and limit access (input, process, storage) Provide for centralized storage with encrypted data Provide for secure transmission of data

Many researchers, however, are not data scientists or computer professionals and have difficulty assessing the risks relating to electronic data capture, transmission and storage. Additionally, they likely lack the skills to develop software that can securely collect and manage their research data. In some cases, researchers may also lack knowledge of the myriad of laws and regulations that govern how personal information must be protected. Because of this knowledge deficit, researchers may have to consider alternative approaches for meeting complex data capture requirements such as;

- solicit assistance from a colleague to develop code to capture and manage the data
- outsource data collection programming to a third party
- revert to manual processes for data collection and processing
- outsource processing to a cloud computing vendor
- use internal IT resources for data collection and processing on internal hardware

Depending on resource availability (hardware, software, personnel, money, time) and data complexity, researchers may be able to select one or several of the approaches listed above to meet their research data needs. Each alternative, however, may place limits on what can be achieved for a given cost, and a set timeframe. The amount of control over the development process and the ability to respond to changes in the research protocol must also be considered. Lastly, each approach carries a different level of risk by making sensitive research data available to third parties and outside computing environments.

An alternative to these traditional approaches for supporting the research data lifecycle is to use a low-code development platform to mitigate data loss risks, to reduce development costs, and to provide a secure environment for data collection. An overview of low-code development is provided in the next section.

Low-code Development

Forrester defines low-code development platforms as “products and/or cloud services for application development that employ visual, declarative techniques instead of programming and are available to customers at low-or no-cost in money and training time to begin, with costs rising in proportion of the business value of the platforms” (Rymer, 2017, p. 4). This definition is further extended by Gartner to include application platform as a service (aPaaS) which envisions support for the deployment and execution of the application software in the cloud (Vincent, Baker, Natis, Lijima, & Driver, 2017). Another historical perspective is offered by Weiss (2018) who suggests that no-code and low-code development is based on previous trends in IT such as rapid application development (RAD), fourth generation programming languages (4GL) and computer-aided software engineering (CASE) tools. However, he also indicates that the current platforms have evolved to include advancements that support cloud computing and DevOps.

The market for low-code development platforms is large and is forecasted to grow at a Compound Annual Growth Rate (CAGR) of over 44% (Global, 2017). In dollars, the low-code development platform market size is expected to grow from \$4.32 billion in 2017 to \$27.23 billion by 2022. This growth is expected as a result of rising demand for software automation and innovative applications.

Research firms Gartner and Forrester have evaluated a number of organizations that provide aPaas capabilities to enterprises such as Salesforce, Outsystems, Mendix, Kony, Oracle, Appian, and ServiceNow among others. For both firms, Salesforce, Outsystems and Mendix are viewed as the industry leaders, each with several hundred or more customers. Some of the key capabilities offered by these low-code development platforms, include: “a broad range of user-interface, process, data management, reporting and app-management features”, support for public cloud services, security certification, and support for a wide range of use cases. Additionally, the platforms provide dashboards, security permissions, workflow support, and the ability to create application code from design components (Rymer, 2017, Vincent, Baker, Natis, Lijima, & Driver, 2017).

Many of the products identified above also provide for services that facilitate development and operations such as one button deployment, responsive web and mobile apps, built in database, bulk uploading of data, exporting of data in excel for use outside of the system, identity services for authentication, internationalization, deployment domains and SSL certificates, and integration of data with the application. Also, a high productivity development environment is supported through the incorporation of design thinking, agile development/management, model-centric UI designer, business logic modeling, dashboard processing, and test driven development (Rymer, 2017).

Although the target market for several of the comprehensive products has been larger enterprises, the technology has also been utilized by smaller organizations and a number of institutions in higher education including Harvard, Georgia Tech, University of Pennsylvania, Bentley University, Boston College, and Florida Gulf Coast University. At Florida Gulf Coast University, a low-code development environment was utilized in an undergraduate systems analysis and design course and also to develop a research data collection application which will be discussed later in this paper. Users of this technology have ranged from “citizen developers” (McKendrick, 2017) defined as those with no or limited software development expertise to those with several years of experience developing application software. Other users of the technology come from industries such as healthcare, insurance, financial services, manufacturing and non-profits and include organizations such as ING, NHS, Logitech, and Liberty Mutual.

Technically, users of the software can choose between public cloud, private cloud, virtual private cloud and on premises solutions if necessary. These options provide choices, giving users the flexibility to adopt a technical approach that best fits their internal capabilities and fits with their overall computing strategies. The cloud options lessen the need for technical support and provide for faster start-up, development and delivery timeframes. Another key benefit of the software is the availability of components (e.g. pretested apps and frameworks) that can be easily integrated into user solutions. These components provide for specialized routines and business functions that have been previously tested that result in higher quality being built into the end product. An example of one of these components is user authentication during the log in process (Mendix, 2017).

Being able to create an application that runs in the cloud or on-premise computers with little or no coding effort, can help organizations achieve substantial cost, productivity, and quality gains over traditional development approaches. In a benchmark study conducted by QSM (2014) relating to the use of the Mendix App Platform, development effort was estimated to be reduced by 70% and that projects using the technology were shown to require only “1/6 of the development hours compared to the industry average for business application projects” (QSM 2014, p. 10). Other providers of low-code platforms report similar reductions of 75% in development effort and three times faster time to market based on customer feedback (Outsystems, n.d.). The following sections discuss the use of a low-code development platform to create an application to collect data for medical research.

CASE OVERVIEW- HEALTH INFORMATION

Background-Medical Decision Analysis Research

A research project to develop a decision analysis tool in dentistry was conducted that included the development of an Access based tool to serve as a proof of concept and to solicit feedback from the medical community. Subsequently, in order to confirm the validity of the original design and to gather additional insights about factors that influence decision making and patient outcomes, a follow-on project was launched to gather clinical, financial and patient information over a ten-year period from several practitioners. Using the Access model as a starting point, for the data requirements, alternative technology approaches were considered for developing a custom data

collection tool. Given the availability of cloud resources from an education grant, a decision was made to custom develop the application in the cloud to provide practitioner access from around the United States.

An individual with programming experience began the development of the application using a mysql database and forms development software to improve productivity. Over a several month process, a number of key functions were coded (approximately 50% complete), however concerns arose about the quality of the end product and the ability of the researchers to maintain and update the software. As a result, the custom application development to support the research was discontinued.

To expedite the delayed development of the research data collection application, the researcher chose Mendix as the technology to use for the development and deployment. Since the researcher had been evaluating the Mendix Application Platform for use in the classroom for an undergraduate class, the situation presented an opportunity to test the software prior to its introduction and to advance the medical research. The major benefits considered for using the technology to support the research were that it provided:

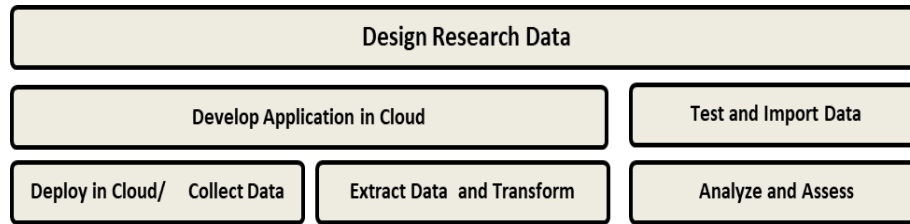
- hands on experience with the platform that can be easily transferred to the classroom
- development productivity (8-10x) based on customer testimonials and benchmarks
- an agile, data driven approach for development
- tested apps to support common functions such as log in/authentication
- hosted processing in the cloud which supported remote secure access and encryption of data
- on-line training for individuals with varying skill level to learn the product
- the ability to “publish” the application to the cloud with minimal effort
- the ability to export and import data from the application
- no up-front cost for development as part of the organization’s University program

A further consideration was input from external assessments of the product from Gartner that described Mendix as “an established (2005) pure-play, low-code hpaPaaS provider offering cloud-native architecture and capabilities. The Mendix Web Modeler together with the Atlas UI framework offers both professional developers and, to a lesser degree, citizen developers interested in high-productivity, a model-driven visual development environment generating metadata that is interpreted at runtime” (Vincent, Baker, Natis, et al., 2017, p.11).

Elements of the Low-code Development Effort

With the re-development of the application based on the “create data” stage of the lifecycle, described in a previous section, the researchers began the process by reevaluating the required data needed for the research. A table by table review led to a decision to “de-identify” data where possible to minimize data loss risk. This de-identification, which eliminated information such as complete birthdays, zip codes, and other identifying information also reduced requirements relating to HIPAA compliance (Strauss, 2017). Capabilities available in the software relating to authentication, secure transmission, and encryption were also reviewed to insure that information entered by practitioners was valid, had referential integrity, and was protected from point of collection through export from the system for analytical purposes. Excluded from the scope of the initial development were the lifecycle stages of “processing” or export of data and the “analysis” which would be performed by other technologies such as SPSS or Tableau. The raw validated data maintained in the Mendix database did however represent a portion of the “preserve” element of the research data lifecycle.

The steps in the “Low-code Development” effort were intuitive and were supported by an initial 7-10 hours of on-line training provided by Mendix. The process for developing and deploying the application is shown in the diagram below and follows a traditional approach. However, a significant difference from traditional development was observed in that the application architecture was provided through the platform, leaving the developer to concentrate primarily on what data is required and how it should be captured.



Once the design of the data requirements was complete, the researcher (developer) began developing the application in the cloud (this work could also have been completed initially on the desktop). The next step in the application development process was to finalize and input the domain model (or data model) that was used in the creation of the database as well as the screens or forms that were used to input, change, delete, and view data. In addition to screens, reports can be created, navigation flows established, data validation rules implemented, and microflows created to handle other processing requirements. Since the application required security so that one practitioner could not see input from another, access rules were also implemented to restrict access to data and to selected forms.

RESULTS

Using the revised data design and the Mendix Application Development platform, a cloud published- working model with test data was operational in approximately a three-week period. No on-site hardware was required other than a laptop with a browser to create the application and users (research participants) could access the system remotely through either mobile devices or smartphones. Users were authenticated against a valid user name and password table and data was securely transmitted from the user to the hosted site operated by Mendix. The database, once established, was accessible to the researchers to parse and export information for analysis and publication. These analysis activities were planned to take place in another system and were not included in the initial development. An added benefit of the platform that was observed is the ability to quickly implement changes to the database and forms in the Mendix Modeler (design tool) as design and research changes are encountered. During the development process, using iterative development techniques, the researchers were able to implement changes with minimal effort.

Identified below are some of the key elements of the system design that were implemented in the research data collection application. The data listed provides an indication of the level of complexity of the application.

Table 4. System Design Elements

Units of Development	Qty
Entities	25
Entities- system admin	10
Forms	25
Logic Flows	3
Roles	2
Validation Table	3
Access Rules	Multiple

SUMMARY

Although detail development time tracking was not performed for either the initial application development process or for the Mendix application development effort, substantial productivity gains were realized using the Mendix technology. These gains could be seen simply by observing the results from the three week development process compared to the several month initiative that did not produce a working system. Other gains were realized in reduced turnaround time for making changes to either the data or process design and subsequently having those changes implemented for inspection by other researchers.

As opposed to coding the application in java or some other language, and having to design and code certain commonly used functions such as authentication, the low-code development approach also greatly simplified not only the creation of the application, but the database as well. Additionally, security and access rules were able to be easily implemented in the application, and the entire process of launching and monitoring an application in the cloud was handled by the architecture. Using an Agile approach to development, revisions and updates to processes and data requirements were also able to be implemented with less effort.

Another consideration for researchers or those considering the use of the Mendix low-code development platform, is that it can be used by those with little or no programming experience. “Citizen” developers can use the platform to create simple applications that collect and manage data for their research or alternatively they can become actively engaged with data design and modeling with others who have better skills/knowledge. Technologies such as Mendix may also offer researchers an opportunity to quickly develop applications to support their research needs, complete their data collection faster, provide a location for the storage of research data, and may help to mitigate risks relating to the safeguarding of data.

For researchers evaluating the use of low-code development platforms, it will be important to understand potential costs for using the platform in production since this appears to be a new area for providers. Also, when implementing these solutions, researchers must understand where the data is being stored, if back-up copies are being made, and how they are able to input or export data from the system to minimize the potential loss or exposure of data.

An area of further research for the use of low-code development platforms include assessing how effective “citizen” developers can be using the technology since research is conducted in many disciplines. This is particularly important when programming skills are not a core competency. Additionally, exploration of how this technology may or may not support other components of the research data lifecycle would be helpful for researchers to understand as they consider the use of the platform.

REFERENCES

- Data [Internet]. Oxford, U.K.: Oxford University Press [2018] Retrieved from <https://en.oxforddictionaries.com/definition/us/data>
- FGCU. (2017). *IRB guidelines*. Fort Myers, FL: Florida Gulf Coast University. Retrieved from https://www2.fgcu.edu/ORGS/Files/IRB_Guidelines.pdf
- Global low-code development platform market: Rising demand for innovative applications and software automation to enable market exhibit strong growth. (2017, Nov 29, 2017). *NASDAQ OMX's News Release Distribution Channel*, pp. n/a. Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1969829335?accountid=10919>
- Federal policy for the protection of human subjects, 45 CFR part 46 (1991).
- Federal Register, (2017). Volume 82 Issue 12
- How to quantify the ROI of low-code application development: Webinar hosted by OutSystems and Nucleus Research. (2016, Nov 17, 2016). *Business Wire*, pp. n/a. Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1840746064?accountid=10919>
- Marie-Line Gentil, Cuggia, M., Fiquet, L., Hagenbourger, C., Thomas Le Berre, Banatre, A., Chapron, A. (2017). Factors influencing the development of primary care data collection projects from electronic health records: A systematic review of the literature. *BMC Medical Informatics and Decision Making*, 17, n/a. doi:<http://dx.doi.org/10.1186/s12911-017-0538-x>
- Mckendrick, J. (2017). *The rise of the empowered citizen developer*. (). New Providence, NJ: Unisphere Research.

Issues in Information Systems

Volume 19, Issue 2, pp. 132-139, 2018

- Mendix closes enterprise UX gap with atlas. (2017, Dec 5, 2017). *PR Newswire* Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1972266552?accountid=10919>
- Mendix named a leader in gartner's 2017 magic quadrant for enterprise high productivity application platform as a service. (2017, May 1, 2017). *PR Newswire*, pp. n/a. Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1893584238?accountid=10919>
- OutSystems boldly cuts mobile application development time in half with low-code platform; OutSystems 10 is the first solution to combine the speed of a low-code platform with advanced mobile app development. (2016, Oct 4, 2016). *M2 Presswire*, pp. n/a. Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1825438369?accountid=10919>
- OutSystems helps NHS PHP put the patient at the center of care using low-code application development. (2017, Jul 26, 2017). *Business Wire*, pp. n/a. Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1923282485?accountid=10919>
- Press release: Mendix announces global reseller agreement with SAP, providing enterprises low-code development for building web and mobile apps at speed and scale. (2017, Sep 26, 2017). *Dow Jones Institutional News* Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1942983980?accountid=10919>
- QSM. (2014). *Independent benchmark study*. (Electronic). Quantitative Software Management.
- Rymer, J. (2017). *The forrester wave: Low-code development platforms for AD&D pros, Q3 2017*. (). Cambridge, MA: Forrester Research.
- Strauss, L. J. (2017). De-identifying protected health information. *Journal of Health Care Compliance*, 19(6), 51-54.
- Survey finds 74 percent of IT professionals see low-code solutions as key to rapid application development. (2017, Apr 11, 2017). *PR Newswire*, pp. n/a. Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1886263908?accountid=10919>
- Survey: Low-code platforms solve enterprise need for speed across multiple digital initiatives. (2017, Aug 2, 2017). *NASDAQ OMX's News Release Distribution Channel*, pp. n/a. Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/1925144307?accountid=10919>
- Summary of HIPAA security rule, (2013). Retrieved from <https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html>
- UK data archive. create and manage data: Research data LifeCycle, UK: University of Essex. (2018). Retrieved from <http://www.data-archive.ac.uk/create-manage/life-cycle>
- University of Edinburgh information services. research data management programme: Research data management home. (2018). Retrieved from <http://www.ed.ac.uk/schools-departments/information-services/research-support/data-management/data-management-home>
- Vincent, P., Baker, V., Natis, Y., Lijima, K., & Driver, M. (2017). *Magic quadrant for enterprise high-productivity application platform as a service*. (No. ID: G00304071). Online: Gartner.
- Weiss, T. R. (2018, Apr 20, 2018). No-code, low-code development platforms help organizations meet growing app demand. *SQL Server Pro*, Retrieved from <http://ezproxy.fgcu.edu/login?url=https://search.proquest.com/docview/2027691460?accountid=10919>
- Whitman, M. E., & Mattord, H. J. (2017). *Management of information security* (Fifth ed.). Boston, MA: Cengage.