C3.ai

Best Practices in Enterprise AI Application Development

A proven methodology for designing, building, and deploying Enterprise AI applications at scale

Table of Contents

 © 2020 C3.ai, Inc. All Rights Reserved.

Overview

Enterprise AI is a powerful new category of software that is a core enabler of digital transformation. All large organizations will deploy dozens – perhaps hundreds, or even thousands – of AI-enabled software applications across every aspect of their operations. These applications leverage 21st century technologies – including elastic cloud computing, big data, the internet of things, and advanced methods of artificial intelligence – to address a broad and growing range of use cases.

Despite the potential of these technologies to drive significant value, many organizations are failing in their digital transformation efforts. According to McKinsey & Company, fewer than 20% of enterprises achieve sustained value from their transformation initiatives.[*]

In particular, many organizations are struggling to develop and deploy this new class of AI application software. Compared to earlier generations of enterprise software such as CRM or ERP, the requirements for developing enterprise AI applications and deploying them at scale can be daunting. AI applications involve ingesting, aggregating, and processing massive volumes of disparate data (structured and unstructured), from numerous sources. AI applications also require the development, training, and tuning of machine learning models – in some cases hundreds, thousands, or more for a single application.

These requirements can pose significant challenges for even highly skilled and experienced IT organizations. Enterprises that succeed at AI follow a set of best practices for implementing this new class of enterprise software. Based on experience in working on some of the largest and most sophisticated AI implementations globally – at 3M, Shell, Enel, ENGIE, US Department of Defense, and others – C3.ai has developed a complete, end-to-end application development methodology that codifies these best practices.

Organizations across multiple industries are applying this methodology to develop and deploy AI-enabled applications at industrial scale for numerous high-value use cases: predictive maintenance, inventory optimization, energy management, cash management, production optimization, fraud detection, and many others.

This document provides an overview of this methodology. It summarizes best practices for designing, developing, testing, deploying, and operating enterprise-class AI applications at scale, outlining the key phases and activities, from use case conception through application development to production. By following these best practices and prescribed methodology, organizations can minimize risk and significantly increase their likelihood of successfully deploying enterprise AI applications to achieve meaningful and sustained results.

[*] Unlocking success in digital transformations," McKinsey & Company, October 2018

© 2020 C3.ai, Inc. All Rights Reserved.

Enterprise AI Application Development Methodology

Figure 1 illustrates the overarching activities and governance supporting individual application delivery. Each of the four key phases – Plan, Specify, Build, Operate – encompasses a set of critical activities and outputs.

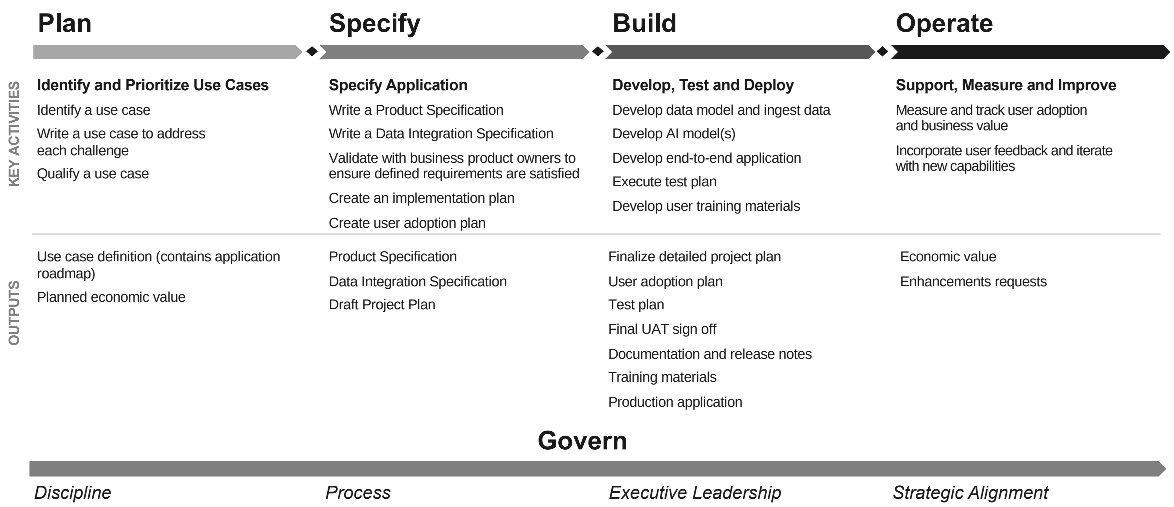

Figure 1: Application development methodology high-level activities

Plan: Ideate a Use Case and Build a Roadmap

The application development lifecycle starts with **identification** of a business problem and a potential solution (together these constitute a *use case*). Use cases are then defined, qualified, and prioritized on an application roadmap.

A clear use case **definition** enables both business and technical leaders to evaluate and qualify a potential application. It aligns business expectations at the outset of the project and establishes a value model for the application that is tracked from initial development through ongoing operations. Alignment between the business and technical development teams ensures that the resulting application adds value and fulfills the use case requirements.

The **qualification** step ensures that a use case is both solvable and worth solving. There are many use cases that can be derived and implemented at small scale. However, when scaling to the enterprise, the required data availability or ability to transition into the business operations may be limited or impractical.

A good use case identifies:

- Business Value: What is the quantified projected value of the application?
- AI Addressability: Is the use case suitable to be solved using AI techniques?
- Ease of Implementation: What are the technical, resource, and change management requirements?

 © 2020 C3.ai, Inc. All Rights Reserved.

- Time to Value: A good business case can see ROI within 6-12 months.

Managing a digital transformation requires **developing a rubric** to prioritize and greenlight new application development. In addition to the elements identified in the business case, the decision-making rubric includes a consideration of the organization's maturity in implementing AI-based applications as well as user adoption: If the users will not accept the recommendations generated by the application and take action, the value will not be realized.

The point of the rubric is to create a rigorous process to evaluate one use case against another in situations where you have identified multiple legitimate use cases.

Specify the Application

Once an economic business case has been approved for development, a detailed technical **product specification** is built. This is a joint effort of the project team, led by the business product owner assisted by a solution architect, application developers, and data scientists. The specification provides the blueprint for all requirements, machine learning (ML) models, and user experience. The specification drives the estimate for the workload and creates an understanding with the business about timelines for production release and ROI.

Identifying the underlying data model for the application is key to writing the full detailed specification. This includes identifying the sources of the data, the profile of the data, the quality of the data, and most importantly how to interpret the data. The ML model developed for the application is highly dependent on this data.

The product specification includes two main parts. The first part is a written document that outlines:

- User Interface, describing how a user would interact with the application;
- Data Model, defining an object-oriented representation of the data and their relationships;
- AI, specifying whether this is predictive or detective in nature, and how it provides actionable insights; and
- Application Logic, detailing the end-to-end workflow to facilitate the integration of AI insights into a business process.

It is critically important to have a product owner involved from the beginning on the product specification in order to drive this process and gain alignment on all parts.

The second part of the product specification outlines the **integration architecture**. New data need to be consistently loaded into the application in order to create new recommendations or generate updated results. Designing the architecture to integrate relevant source systems is a critical phase in the overall process. Three steps are involved in defining the integration architecture:

1. Identify and define the service level agreement (SLA) – The application will be architected differently depending on what it is being optimized for, whether volume and throughput, low latency, or cost.

© 2020 C3.ai, Inc. All Rights Reserved.

2. Define the infrastructure architecture – Whenever possible, key infrastructure-level decisions should be made uniformly across all source systems in order to reduce complexity.
3. Document the enterprise systems, sensors, and/or third-party data providers in scope – Avoid bringing in everything “just in case.” This increases complexity, cost, and consequently time-to-value of the solution.

Build the Application

The build phase encompasses application development, testing and tuning, and release.

Development

There are three main work streams in the development phase. These can be largely parallelized to accelerate development.

- **Data integration development** involves building and/or configuring extraction and transformation mechanisms to bring data from many sources, in many formats, into the application. This development work often requires an iterative build-and-test process due to common complexities in change data capture, data transformation, and canonical format.

- **Model development and tuning** is the process by which a data scientist develops and tunes a machine learning, deep learning, or other analytical model on the assembled, normalized, and correlated data set. This work includes both feature engineering and tuning as well as configuration of a model pipeline that will eventually enable production automation. Results of these models should be shared regularly with subject matter experts and the project team to ensure progress is aligned with the project goals.

- **Application logic and user interface development** involves building application logic required to integrate AI-generated insights into a current (or future-state) business process. Most applications also include a web-based user interface through which end users interact with the application.

Testing and tuning

The quality of an AI-based application is validated across four dimensions:

- **Functional** – Do the application features perform as expected and documented in the specification?
- **ML model validation** – Does the model produce actionable insights (insights relative to an established baseline or minimum)?
- **Performance, scalability, and reliability** – Does the application meet all SLAs at production scale?
- **User acceptance** – Does the application operate in a way deemed acceptable to end users?

Each of the four QA cycles above should be carefully scoped with test plans, managed with detailed timelines, and, upon successful completion, signed off on by respective owners. Only after all QA cycles are complete can an application be released into production.

 © 2020 C3.ai, Inc. All Rights Reserved.

ML model validation comprises two components: validation of the results and tuning of the model. As the results are validated with the business, the data scientists will perform final tuning on the model's features and hyperparameters.

Release

During development, a **launch plan** is defined. The launch plan includes training for end users and a user adoption plan. Updated business processes are described based on the AI-enabled insights and user adoption KPIs are defined. User adoption is paramount to the success of the overall application and end users need to buy in to the recommendations. The **user adoption plan** includes comprehensive end user training, so end users understand how to interact with the application to do their jobs.

Production deployment occurs after all features have been fully developed and passed all quality assurance validation. Production deployments are scheduled to minimize impact on end users and are coordinated with the business stakeholders. The development team should create a checklist for deployment in order to ensure that all steps are followed and executed in the right order. For all new application deployments, users should be trained just in time for the deployment.

Typical Application Development Flow

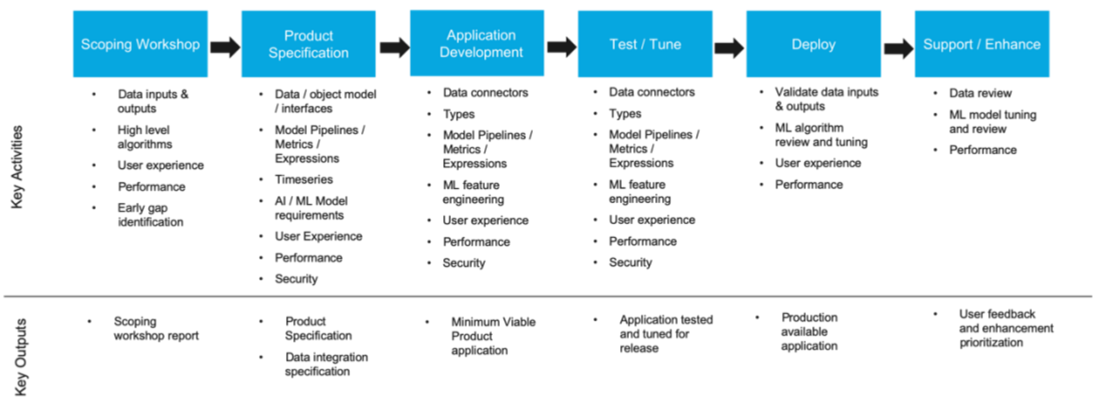

Figure 2: Best Practice Application Development Flow

Operate the Application

Successfully operating the AI-enabled application requires a clearly defined **support process** including an escalation path for complex issues or product issues. **Continuous measurement and improvement** are a critical part of ongoing operations. There are two main components to consider:

- Managing and enhancing ML models: ML models are continuously improved as the business evolves, new data become available, and users provide feedback. This work may include performance improvement, retraining, feature set enhancement, explainability enhancement, champion-challenger testing and configuration, and numerous other efforts.

 © 2020 C3.ai, Inc. All Rights Reserved.

- Gathering user feedback: User feedback is continuously solicited in production. Users should have well-defined avenues for submission of improvement and new feature requests of any size. These requests should be funneled back into the centralized prioritization process and considered for future releases by the responsible Product Owner.

Conclusion

The AI application development methodology summarized in this document provides a proven, step-by-step framework to successfully develop AI-based applications. This approach is fully elaborated in the C3.ai Application Development Methodology™ – a comprehensive reference for developing enterprise AI applications on the C3 AI Suite™. The C3.ai Application Development Methodology articulates each phase, providing a detailed set of activities and templates in order to complete the necessary steps to go from an idea to a production application.

To learn more about the C3.ai Application Development Methodology, contact C3.ai at C3.ai/get-started.

About C3.ai

C3.ai is a leading AI software provider for accelerating digital transformation. C3.ai delivers the C3 AI Suite™ for developing, deploying, and operating large-scale AI, predictive analytics, and IoT applications in addition to an increasingly broad portfolio of turn-key AI applications. The core of the C3.ai offering is a revolutionary, model-driven AI architecture that dramatically enhances data science and application development.

Proven Results in 8-12 Weeks | **Visit c3.ai/get-started**

© 2020 C3.ai, Inc. All Rights Reserved.