



# Controls and Novelty on Digital Platforms: Two Case Studies

Onkar Malgonde<sup>1</sup> and Alan Hevner<sup>2</sup>

<sup>1</sup>Operations Management and Information Systems, College of Business, Northern Illinois  
University, DeKalb, IL

<sup>2</sup>School of Information Systems and Management, Muma College of Business, University of  
South Florida, Tampa, FL

omalgonde@niu.edu, ahevner@usf.edu

## Abstract

Software system projects face challenges to rapidly meet user requirements while adding novel values to the application domain. Value appropriation focuses on exploiting existing knowledge to develop software that meets market requirements. Value creation focuses on exploring the solution space to innovate and attract new customers. In this pilot research, we study the tension between software project controls and the goal of novelty in the software product. Two case studies provide preliminary evidence that a well-balanced portfolio of controls can result in the design and implementation of novel product features. We position the case studies in the context of digital platforms to bound our definitions of control mechanisms and novelty. Based on analysis of data collected from two case studies, we find that formal and informal control modes can positively influence novelty in software applications on digital platforms. We conclude with a discussion on implications for software development and future research directions.

## 1 Introduction

Recent advancements in software systems and information technologies have catalyzed digital transformation initiatives within organizations and renewed organizational focus on innovation [1, 2]. To support such endeavors, we are witnessing changes in business practices such as partner networks, subscription-based usage, and consulting partners. To support the renewed focus on innovation in digital era, software development projects are increasingly identifying and incorporating enabling technologies and tools such as platform-based application development, low-code development platforms, enterprise application development packages, and prototyping tools [2].

This research focuses on balancing value appropriation requirements and value creation opportunities in the context of digital platforms which represent a mainstream channel for development and deployment of software development projects. Software application development on digital

platforms require project teams to achieve application-platform match, application-market match, exceed core value proposition of the platform, and novelty of the application, in addition to traditional outcomes of IS project success such as project efficiency, quality, and adaptiveness [3, 4, 5].

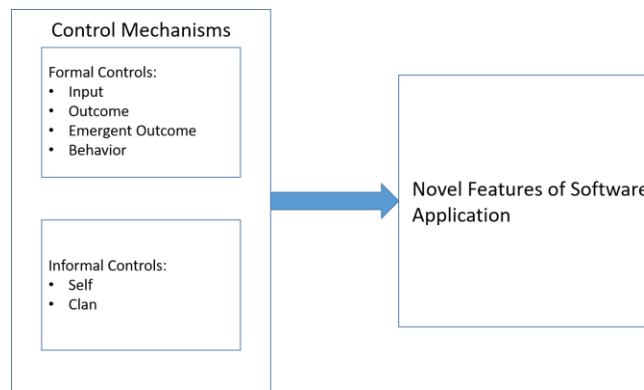
A key research question is how to achieve the right balance between project controls while supporting the creative design of novel features. Prior research has found contradictory results. Several studies focusing on the control of software projects identify a negative effect of control mechanisms on innovation outcomes [6, 7] suggesting a “stifling of creativity and limiting of adaptability” [6, p. 225]. However, many software projects effectively release novel applications on different platforms and in many application domains. Thus, we explore the following research question:

*How do formal and informal control mechanisms relate to the development of novel features in software projects on digital platforms?*

To address this question, we conduct two case studies. Via qualitative interviews with members of the two software development teams, we identify the control mechanisms applied and the novel features created and study the relationships between the two. We organize this paper as follows. In section 2, we discuss our research model. In section 3, we discuss our research methodology, data collection, and analysis approach. We present the criteria for selecting cases (two case studies). Section 4 discusses the results of each case study. We conclude in section 5 with the implications of this research for software development project controls and propose future research directions.

## 2 Related Literature and Research Model

Controls in software projects have received considerable attention in research and practice [5]. However, there exists a lack of rigorous study on the effect of control mechanisms on novelty of software application features [5, 8]. This focus leads us to our research model as seen in Figure 1.



**Figure 1. Controls and Novelty Research Model**

### 2.1 Control Mechanisms

Software development controls, the independent variable of the study, are of two types; formal and informal. Formal control types are classified as input control, behavior control, outcome control, and emergent outcome control. The key distinction for formal control is the presence of a controller and a controllee. Each type of formal control is briefly described.

- **Input Control:** The controller assigns resources (inputs) to the development project that are sufficient to successfully complete a desired result. There is limited discussion on the use of input control for software projects [3]. In the context of R&D projects in the pharmaceutical industry, Cardinal [7] finds empirical evidence in support of input control leading to incremental and radical innovation. Software project managers can alter resources such as team composition, technical environments, tools, and knowledge resources, among others, for the project team to facilitate identification and assimilation of novel features for the focal application.
- **Behavior Control:** The controller uses processes and rules to direct controllees towards accomplishment of organizational goals. In software project control, behavior control is exercised using mandated routines such as meetings and development methodologies that signal use of specific methods in the project. Prior research attributes use of behavior control to outcomes of project adaptiveness, efficiency, and quality [3]. In the context of R&D projects in the pharmaceutical industry, Cardinal [7] find a negative effect of behavior controls. Some studies posit a negative effect of behavior control due to rigidity and lack of experimentation which stems from specifying controllee behavior [4, 6].
- **Outcome Control:** The controller specifies an outcome and evaluates the project based on achieving the outcome. For outcome control to be effective, the controller should be able to specify the outcome a priori and measure the outcome. Typically, software specifications and timelines form popular control mechanisms for outcome control. In the digital platform environment, novelty of an application needs to change over time, as platform and competitors update their offerings. Also, measuring novelty of an application is challenging [8].
- **Emergent Outcome Controls:** The use of scope boundaries and ongoing feedback are important forms of emergent outcome controls that allow the project team to revise outcomes that are difficult to identify a priori by facilitating feedback [9]. Scope boundaries channel the team's efforts while allowing autonomy within the boundaries. For development teams developing applications for digital platforms, emergent outcome controls provide mechanisms which can enable the team to explore the technological space provided by the platform and seek feedback from within and outside the IS team.

Informal controls rely on a software team's shared values and vision of the application. Two forms of informal controls are clan and self.

- **Clan Control:** Shared values among team members motivate the project to a successful result. Chua et al. [10] find that clan controls need to be developed over time with careful maneuvering to be effective. Although difficult to implement [11], clan controls demand minimal monitoring once implemented. Empirical evidence suggests a positive effect of clan control on project's success [3], [10]. In digital platform environment, clan control can play an important role in channeling the team's efforts towards developing a novel application [8].
- **Self Control:** Team members have internal motivations to self-direct their actions to achieve project goals. Prior research suggests a positive effect of self-control on project's success [3]. In the dynamic environment of digital platform, it is important to enable individual team members to identify novel features that will set apart the focal application. The project manager may identify appropriate control mechanisms to enable team members to exercise self controls to experiment with features and technological advancements to develop novel features for the application.

## 2.2 Application Novelty

Traditionally, organizations develop innovative product lines through a linear value chain [12]; products are designed, developed, and marketed by a single firm. However, with the pervasive digital innovations

and technology, the locus of organizational innovation has shifted to digital software platforms which rely on external entities to develop innovative solutions. Thus, current conceptualizations of novelty in an application refer to the features and extensions offered by the application relative to the platform and other competing applications [8]. From our analysis, we extend this definition of novelty, as the dependent variable of our study, to include content provided by the application, data sources and its design, user interface, alerts/messages, and platform's ecosystem that distinguishes the focal application from its competition (competing applications that may or may not be on the same platform). Consequently, novelty of the application is not limited to its features. Novelty for a focal application may arise from its choice of platform since the application's user may not differentiate\* between the application and its platform.

## 3 Research Methodology

### 3.1 Research Study Design

This research studies the relationship of control mechanisms and the novelty of the platform-based application. We conduct case studies to answer our research question. A case study methodology is appropriate when 'how' questions are posed in the research [13]. Case studies also allow us to extract a nuanced understanding of the control mechanisms identified by software development teams that contribute to novelty of the application. The unit of analysis is the IS project team that is developing the application on digital platform. To provide empirical grounding, we conduct two case studies.

### 3.2 Case Selection and Site Description

Selection of projects is driven by following criteria: (a) the application is built on a digital platform, (b) stakeholder roles (controllers and controlees) can be identified, (c) competing applications for the focal application exist, (d) novel features in the focal application are identifiable, and (e) ownership of application and platform are not held by same organization(s). Table 1 summarizes and Malgonde [14] provides details about the case study sites, number of interviews at each site, informant roles in the interviews, and provides a brief description the projects at the case study sites.

### 3.3 Data Collection

To test the efficacy of our selection criteria, interview protocols, and theoretical understandings, we performed two pilots [14]. Our first pilot location is an IT-department of a large public university in the Southeastern United States. The application under consideration allows universities to complete their reporting obligations for state-mandated requirements. The second pilot location is a Fortune-500 organization with a large development team, analysts, architects, and a project manager. This team is developing an application that supports online subscription of enterprise software. The application connects to multiple external platforms, increasing the complexity of the project. Interestingly, the application serves as a platform to other connected applications.

The pilots refined our interview protocols and data analysis methods. The interview protocol forms the basis for each interview with occasional deviations to accommodate any contemporary issues such as reordering questions based on an interviewee's response or dropping certain questions that are not consistent with interviewee's role in the project. Follow up questions (not included in the protocol) may be included to seek clarification and/or reconfirmation. Finally, questions exploring interviewee's role

---

\* In enterprise grade applications, users are often unaware about the digital platform and its offerings when using the application.

in the development project may be included to better understand the controls adopted by the team. Table 1 provides a summary of the two case study sites, named AT and TB. In case of AT, majority of interviews took place on-site whereas a small fraction are individual online sessions. In case of TB, all the 7 interviews are individual online sessions.

Organization	Interviews	Informant Roles	Project Description
AT is a software consulting firm specializing in development, maintenance, and deployment of software applications across different industries.	9	<ul style="list-style-type: none"> <li>• Delivery Lead</li> <li>• Team Lead</li> <li>• Senior Developers</li> <li>• UI Designer</li> <li>• UI Developer</li> <li>• Technical Architect</li> </ul> 24.5	The client (a non-profit organization) wants to develop a mobile application (iOS-based) that would allow healthcare professionals to stream on-demand educational content, videos, support dynamic notetaking, and resume playback.
TB develops IT solutions to challenges in different domains such as CRM, Healthcare, and Operations	7	<ul style="list-style-type: none"> <li>• Product Owner</li> <li>• Product Manager</li> <li>• Practice Manager</li> <li>• Sales Consultant</li> <li>• Technical Architect</li> <li>• Solution Architect</li> <li>• Functional Consultant</li> </ul>	The product is a cloud-based Healthcare management application and competes with other offerings on Microsoft Azure platform. TB partners with select customers (hospitals) to develop features which are incorporated into the product – streamline patient care with CRM platform and consolidate patient care.

**Table 1. Summary of Case Study Sites**

### 3.4 Data Analysis

To answer our research questions, we analyzed the qualitative data in two phases [15]. First, open coding systematically labeled our data to identify events, actions, and interactions. Second, axial coding related categories and subcategories around the data. We used independent coders with experience in software development projects to code our data. Coders were trained using the pilot interviews’ transcripts. We discussed and clarified any ambiguity in conceptual understanding and operational definitions. Coders were blind to the research question. We performed within case analysis, followed by cross-case analysis. Our analysis focused on identifying key findings on the relationships between control mechanisms and the novel features of the software system under development.

## 4 Results

### 4.1 Case Study 1: AT

#### 4.1.1 Control Mechanisms

We find use of different technical environments to facilitate experimentation with new ideas. Specifically, teams use *sandbox* environments to trial new ideas, *demo* environments to integrate new ideas with existing application, and *quality assurance* environments to test new ideas with existing application features; in addition to the *production* environment that hosts the actual application. Setting, maintaining, and transferring artifacts from such environments utilizes resources (time and cost). In case of AT, we find controls in the form of changes in team composition to facilitate development of novel features. Specifically, the video playback feature with positional saving and scrolling for AT’s

application is not supported by the platform’s native capabilities and requires technical expertise. We find use of *collaborative sessions* aided by interactive mockups and designs to be a key behavioral control mechanism to identify and refine novel features for the application. We find different labels for these sessions: backlog refining/grooming (AT), or brainstorming sessions (including a session with the project manager of a Fortune 500 organization). We find support in the use of ongoing feedback during development of the application. The traditional conceptualization of ongoing feedback has a directionality from clients/users to the development team. However, we find that feedback may be bidirectional during and/or after iterations. Table 2 summarizes formal control mechanisms in AT that contribute to the novelty of the application.

<b>Control Mode</b>	<b>Control Mechanisms</b>
Input Controls	<ul style="list-style-type: none"> <li>• Change team composition (add and drop skills/personnel) to identify and develop novel features</li> <li>• Setup different technical environments to execute proof of concepts and integrate ideas in the application</li> </ul>
Behavior Controls	<ul style="list-style-type: none"> <li>• Facilitate workshops with users/clients at intervals</li> <li>• Interactions/feedback loops during iteration demo</li> <li>• Collaborative discussions/workshops between IS team and users with user interface mockups</li> <li>• Use technical capabilities to identify new features</li> <li>• Discuss technical approaches to achieve new features</li> </ul>
Output Controls	No evidence found
Emergent Outcome Controls	<ul style="list-style-type: none"> <li>• Feedback during and after each iteration</li> <li>• Ongoing feedback from application’s usage data</li> <li>• Content and how it is served by the application</li> </ul>

**Table 2. Formal Control Mechanisms in AT project**

<b>Control Mode</b>	<b>Control Mechanisms</b>
Clan Controls	<ul style="list-style-type: none"> <li>• Shared understanding on success criteria</li> <li>• Prefer physical presence for client meetings whenever possible</li> </ul>
Self-Controls	No evidence found

**Table 3. Informal Control Mechanisms in AT project**

We find evidence for the use of clan control in the AT project. Discussions with the client’s liaisons pertaining to requirements are typically viewed as meetings that facilitate the project’s understanding of the domain. However, as part of these discussions, the team discusses alternative ways to either improve known requirements or recommend new requirements based on the team’s prior experience. In case of AT, we see evidence of a clan mentality where entrepreneurial thinking [8] is encouraged so that the client is successful. However, we also see the negative influence of a “consultant” mentality. Specifically, AT’s consultants realize that their role is limited to a module. Similarly, AT’s leadership team acknowledges that decisions on features are made by the client based on time, cost, and desirability of recommended feature. Such client-vendor relationship may hinder identification and assimilation of novel features as AT may be wary to discuss potential features. Table 3 summarizes informal control mechanisms for AT project.

#### 4.1.2 Novel Features

Novel features in AT project are threefold. First, the application allows its users to seamlessly stream content such as text, pictures, and videos on mobile devices. This requires dynamic adjustments to the content. The content is stored on the client’s servers. Previously the native video playback feature from

the platform lacked finesse. Second, the application enables users to make and retrieve notes while they are watching videos. Finally, the user can resume video playback from the last viewed location. These features set the focal application apart from competing applications on the platform and off-platform alternatives including client’s website. Some of these features exist in other domains. For example, resuming video playback from the last view position has been staple for video streaming application on the same platform. However, the ability to stream video content and extend platform’s native capability is novel in the client’s competition space.

<b>Novel Features</b>
<ul style="list-style-type: none"> <li>• Allow users to seamlessly stream content such as text, images, and videos, on mobile devices. This requires dynamic adjustments to the content. The content is stored on the client’s servers. The native video playback feature from the platform lacked finesse.</li> </ul>
<ul style="list-style-type: none"> <li>• Application enables users to make and retrieve notes while they are watching videos.</li> </ul>
<ul style="list-style-type: none"> <li>• User can resume video playback from the last viewed location.</li> </ul>

**Table 4. Novel Features of AT application**

## 4.2 Case Study 2: TB

### 4.2.1 Control Mechanisms

At TB, the management team identified *springboard* clients (“early adopters” as noted by TB’s Product Manager) that provided market needs and differentiators in exchange for access to the application. We classify partnerships with such clients as input control due to the emphasis by controller to leverage access to springboard clients and temporary association with such clients. One of TB’s team members (title—Solution Consultant) is a registered nurse (the target users of TB’s application) who participates in product demonstrations and identification and vetting of potential features. Inclusion of a team member who can provide users’ perspective is another input control mechanism. TB teams use mock-up screens and designs so that all stakeholders can visualize potential features, alter designs to visualize focal feature, and identify approaches to incorporate potential features in the application. In addition to discussions based on interactive mockups,

TB’s team also performs configurational changes to the existing application and platform, where possible, to visualize new ideas for the application. We find identification of technology and tools to be made upfront which determines the scope boundaries for technical exploration. In case of TB, there exist organizational restrictions on the use of technology and tools provided by a vendor. We also find that choice of the digital platform introduces a major scope boundary for the team as technical capabilities and tools are bounded. The project team must identify alternatives that can be supported by resources within the scope boundary. We also find use of feedback mechanisms, part of emergent outcome control mode, to facilitate an individual’s exploration of specific ideas. Table 5 summarizes formal control mechanisms in TB project.

In case of TB, we find a shared understanding and importance of novelty across team members. The shared success criteria encourage the team to identify and vet alternatives. Also, we find that team members appreciate and recognize the value contributed by other team members. For example, analysts recognize the possibilities and limitations faced by the technical team in implementing proposed features. To address limitations, analysts identify feasible alternatives and discuss with the technical team. Further, we find common consensus on the importance of certain processes and events. For example, meetings and discussions with potential customers is an important opportunity to verify and gather new feature ideas. For self controls, proactive team members may experiment to identify novel features for the application. After implementation, typically as a proof-of-concept, other team members

with closer market knowledge may adapt and integrate the novel feature. Given the shared understanding and importance on identification of novel features, the team is willing to discuss and improve any ideas put forth. Table 6 summarizes informal control mechanisms in TB project.

<b>Control Mode</b>	<b>Control Mechanisms</b>
Input Controls	<ul style="list-style-type: none"> <li>• Identify and partner with springboard clients</li> <li>• Choice of focal platform—platform’s features, ecosystem, and maturity—to distinguish the application from competition</li> <li>• Application’s user as part of the IS team</li> <li>• Technical members attend seminars and conferences hosted by the platform</li> <li>• Setup different technical environments to execute proof of concepts and integrate ideas in the application</li> </ul>
Behavior Controls	<ul style="list-style-type: none"> <li>• Configurational changes to platform before discussing features</li> <li>• Interactions/feedback loops during experimentation, testing, and documentation (within IS team)</li> <li>• Interactions/feedback loops with springboard clients</li> <li>• Collaborative discussions/workshops between IS team and users with user interface mockups</li> <li>• Use technical capabilities to identify new features</li> <li>• Discuss technical approaches to achieve new features</li> </ul>
Output Controls	No evidence found
Emergent Outcome Controls	<ul style="list-style-type: none"> <li>• Application is always ready for demo and feedback</li> <li>• Ongoing feedback from application’s usage data</li> <li>• Content and how it is served by the application</li> </ul>

**Table 5. Formal Control Mechanisms in TB project**

<b>Control Mode</b>	<b>Control Mechanisms</b>
Clan Control	<ul style="list-style-type: none"> <li>• Shared understanding on the importance of processes (for example, team visit to springboard client’s site) to identify novel features</li> <li>• Challenge team members to extend existing feature set</li> </ul>
Self-Control	<ul style="list-style-type: none"> <li>• Proactive team members try new ideas and discuss alternatives</li> </ul>

**Table 6. Informal Control Mechanisms in TB project**

4.2.2 Novel Features

<b>Novel Features</b>
<ul style="list-style-type: none"> <li>• Data management in the application is patient-centric whereas competitors use an event-centric approach. With a patient-centric approach, application’s users can view all records for a patient on the dashboard.</li> </ul>
<ul style="list-style-type: none"> <li>• User experience is highly rated. This includes the application’s ease of use and performance.</li> </ul>
<ul style="list-style-type: none"> <li>• Easy integration with Microsoft’s productivity suite that may be already functional at client’s location.</li> </ul>

**Table 7. Novel features of TB application**

Novel features for TB’s application are threefold. First, data management in the application is patient-centric whereas competitors use an event-centric approach. With a patient-centric approach,



application’s users can view all records for a patient on the dashboard. Second, user experience is highly rated. This includes the application’s ease of use and performance. Third, easy integration with Microsoft’s productivity suite that may be already functional at client’s location. Another area of novelty for TB’s team is the choice of platform. Microsoft’s Azure platform integrates with Microsoft’s productivity suite such as calendar, business intelligence reports, emails, and so on, allows TB’s application to differentiate itself from its competitors that use different platform.

### 4.3 AT and TB Case Study Findings

Summarizing AT and TB case study findings, we find compelling use of formal and informal control mechanisms that lead to identification and assimilation of novel features which exceed the platform’s core proposition and/or differentiate the application from its competition. Specifically, we find use of a mixed control portfolio that consists of formal and informal controls. We do not find support for traditional outcome control mechanisms which we attribute to challenges in specifying outcome a priori in the dynamic digital platform environment. Finally, we find positive influence of a more long-term orientation of the team (TB case) in comparison to a short-term focus (AT case) which can be attributed to the perpetual mode of application under development in platform environments.

## 5 Discussion and Future Research Directions

In this research, we identify control mechanisms and novel application features on digital platforms. Based on analyses of two case studies, we find that formal control modes and informal control modes can contribute to novelty. Finally, changing dynamics of digital platforms and market requires a long-term orientation and consistent identification and assimilation of novel features.

In our case study research, we find empirical evidence in support of platform’s role to enable both formal and informal controls. The role of a project manager to build a mixed portfolio of controls to maximize team members’ contributions to application novelty is a key finding of this research. Platforms are a common point of reference during discussions, decision-making, and collaborative sessions. For example, as teams identify new features, a major focus is how to implement the potential feature using the platform’s current capabilities. The digital platform also facilitates high control communicational and evaluation congruence. The platform plays the role of an anchor that is referenced to alter the control portfolio as the project evolves and to evaluate the current appropriateness of the portfolio. As new and/or updated platform offerings are visible to all, antecedents to control portfolio changes are visible and more likely to be accepted by the team.

In an organizational context, other control modes such as structure, market, and culture, have been considered in related domains [7]. However, software systems project control has been theorized to focus on “temporary organizations” that require different control activities [3] than the larger counterparts of organizational control. In our analysis, we find evidence that challenges this notion of controlling a temporary organization. Applications developed on digital platforms may be perpetually in the state of development due to changes in market and platform.

Our study has two major implications for software project research. First, our findings align with related domains where control’s effectiveness to introduce innovation has been established [7]. This finding calls for deeper studies of how controls and novelty relate in software development projects [6]. Second, this research addresses the recent call to investigate project controls in the digital era [2]. Specifically, we isolate the relationships of controls and novelty. Such a purpose-oriented focus allows us to investigate the required balance between value-appropriation concerns and value-creation requirements in the digital era.

There are few important future research directions. First, our study considers only two small development projects with in-house applications. Future research can consider other project settings

such as offshoring, large project teams, different application domains, and so on. Second, our findings are limited to IS projects where novelty is incremental. As we move in the digital era, one of the major challenges for future research on the impacts of controls to novelty is to study projects that focus on radical innovation. Third, we did not explore the effect of platform's type on the project control—AT's platform caters to consumers whereas TB's platform caters to enterprises. Consequently, AT's platform has tight integration with products and services offered by the digital platform whereas TB's platform has tight integration with other platform-based services provided by the owner(s). The platform's coupling has consequences for the project as platforms feature updates, releases, and technology that are dependent on other services.

## References

- [1] Y. Yoo, O. Henfridsson, and K. Lyytinen, "The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research," *Inf. Syst. Res.*, vol. 21, no. 4, pp. 724–735, 2010.
- [2] M. Wiener, M. Mähring, U. Remus, C. Saunders, and W. A. Cram, "Moving IS Project Control Research into the Digital Era : The 'Why' of Control and the Concept of Control Purpose," *Inf. Syst. Res.*, 2019.
- [3] M. Wiener, M. Mähring, U. Remus, and C. Saunders, "Control configuration and control enactment in information systems projects: Review and expanded theoretical framework," *MIS Q.*, vol. 40, no. 3, pp. 741–774, 2016.
- [4] B. Fitzgerald, "Formalized systems development methodologies: a critical perspective," *Inf. Syst. J.*, vol. 6, no. 1, pp. 3–23, 1996.
- [5] W. J. Orlikowski, "Integrated information environment or matrix of control? The contradictory implications of information technology," *Accounting, Manag. Inf. Technol.*, vol. 1, no. 1, pp. 9–42, 1991.
- [6] W. A. Cram, K. Brohman, and R. B. Gallupe, "Information Systems Control: A Review and Framework for Emerging Information Systems Processes," *J. Assoc. Inf. Syst.*, vol. 17, no. 4, pp. 216–266, 2016.
- [7] L. B. Cardinal, "Technological Innovation in the Pharmaceutical Industry: The Use of Organizational Control in Managing Research and Development," *Organ. Sci.*, vol. 12, no. 1, pp. 19–36, 2001.
- [8] A. Hevner and O. Malgonde, "Effectual application development on digital platforms," *Electron. Mark.*, vol. 29, no. 3, pp. 407–421, 2019.
- [9] M. L. Harris, R. W. Collins, and A. R. Hevner, "Control of Flexible Software Development Under Uncertainty," *Inf. Syst. Res.*, vol. 20, no. 3, pp. 400–419, 2009.
- [10] C. E. H. Chua, W. K. Lim, C. Soh, and S. K. Sia, "Enacting Clan Control in Complex IT Projects: A Social Capital Perspective," *MIS Q.*, vol. 36, no. 2, pp. 577–600, 2012.
- [11] K. M. Eisenhardt, "Control: Organizational and Economic Approaches," *Manage. Sci.*, vol. 31, no. 2, pp. 134–149, 1985.
- [12] G. G. Parker, M. W. Van Alstyne, and S. P. Choudary, *Platform Revolution*. W. W. Norton & Company, 2016.
- [13] R. Yin, *Case Study Research: Design and Methods*. SAGE Publications, 2008.
- [14] O. Malgonde, "An Effectual Approach for the development of Novel Applications on Digital Platforms," University of South Florida, 2018.
- [15] M. Miles, A. M. Huberman, and J. Saldana, *Qualitative Data Analysis: A Methods Sourcebook*. SAGE Publications, 2013.