

Developing a Framework for Using Artificial Intelligence to Update and Maintain 4D BIM
Schedules

By

Larin Jaff

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Civil Engineering

Price Faculty of Engineering

Winnipeg

Copyright © 2025 by Larin Jaff

Abstract

Natural language interfaces can transform the construction industry by enhancing accessibility and reducing administrative workload in the day-to-day operations of project teams. This thesis introduces the Voice-Integrated Scheduling Assistant for 4D BIM (VISA4D) tool that integrates speech recognition and Natural Language Processing (NLP) capabilities with Building Information Modeling (BIM) to streamline construction schedule updating and maintenance processes. It accepts voice and text inputs for schedule updates, facilitating real-time integration with Autodesk Navisworks, and eliminates the need for direct access to or advanced knowledge of BIM tools. It also provides visual progress tracking abilities through colour-coded elements within the 4D BIM model for communicating task status updates within the project teams. To demonstrate its capability to enhance schedule updating and maintenance efficiency, the VISA4D tool is implemented in an office building project in Canada and user testing is performed. An overall accuracy of 89% was observed in successfully classifying 71 out of 80 tested construction-specific commands, while the user surveys indicated high usability, with 92% of participants finding VISA4D easy to use and reporting consistent command recognition accuracy. This study advances the existing work on AI-enhanced construction management tools by tackling the challenges associated with their practical implementation in field operations.

Acknowledgements

I would like to thank my advisor, Dr. Gursans Guven Isin, for her unwavering support, insightful guidance, and continuous encouragement throughout every stage of this research. Her expertise and thoughtful feedback have played a crucial role in shaping the thesis direction and quality.

I also extend my sincere thanks to the members of my advisory committee, Dr. Ahmed Shalaby and Mr. Daniel Kazado, for their valuable input, constructive suggestions, and support. Special thanks are due to my colleagues and friends, who provided both academic and moral support during the more challenging phases of the research. Your encouragement and friendship made a significant difference.

I would like to thank Sahej Garg and acknowledge his technical support in the development and programming of VISA4D.

I would also like to thank the students who voluntarily participated in the user testing of the VISA4D tool. Their thoughtful feedback and active engagement were invaluable to the development and refinement of the system.

Finally, I wish to express my heartfelt thanks to my family and friends for their patience, love, and belief in me throughout this journey. Your unwavering support has been my greatest strength.

Dedication

To my beloved family, whose unconditional love and unwavering support have been the foundation of every achievement in my life. Your sacrifices, encouragement, and belief in my potential have carried me through the most challenging moments of this academic journey. Without your unwavering faith in me, this milestone would not have been possible.

To my dear friends, who have made this journey not only bearable but memorable. Your companionship, laughter, and understanding have been sources of strength when the weight of research and writing felt overwhelming. You celebrated my small victories and lifted me up during moments of doubt, reminding me that I was never alone in this pursuit.

To Mana Erfani, my incredible study partner and friend, who has been my constant companion through every high and low of this academic adventure. Your presence has transformed what could have been a solitary struggle into a shared journey of growth and discovery. Through countless study sessions, moments of frustration, and breakthrough insights, you have provided not only intellectual collaboration but also the emotional support and motivation that kept me moving forward. Your friendship has been one of the greatest gifts of this graduate school experience.

This thesis stands as evidence to the power of community, love, and friendship in the pursuit of knowledge. To all of you who have been part of this journey, thank you for believing in me, supporting me, and walking alongside me every step of the way.

Table of Contents

Abstract	i
Acknowledgements	ii
Dedication	iii
List of Tables	vii
List of Figures	viii
Chapter 1: Introduction and Background.....	1
1.1 Background.....	1
1.2 Problem Definition and Research Motivation	2
1.3 Research Questions.....	5
1.4 Proposed Approach.....	5
1.5 Contribution of Research.....	7
Chapter 2: Literature Review	9
2.1 Role of AI in Construction Management.....	9
2.2 Natural Language Processing in BIM and Construction Scheduling	12
2.3 Real-Time Schedule Updating.....	14
Chapter 3: Research Methodology and Implementation	15
3.1 Research Methodology	15
3.1.1 Problem Identification.....	16
3.1.2 Design and Development	17
3.1.3 Implementation	18
3.1.4 Evaluation	19
3.2 System Architecture and Implementation Details	20

3.2.1 User Input layer (gui.py)	22
3.2.2 nlp_processor.py	25
3.2.3 Task_manager.py	27
3.2.4 Navisworks_api.py	28
3.2.5 NavisworksServer.cs	29
3.2.6 Navisworks TimeLiner Custom Plugin (Addin.cs)	30
3.3 Use Case Scenarios.....	31
3.3.1 Office Building Construction	31
3.3.2 Educational Building Construction	36
3.4 Validation and Verification of the VISA4D Tool.....	41
3.4.1 Validation	41
3.4.2 Verification.....	44
Chapter 4: Discussion	51
4.1 Discussion.....	51
4.2 Limitations.....	52
4.2.1 Sample Size and Participant Demographics.....	52
4.2.2 Speech Recognition Challenges	53
4.2.3 Technical Infrastructure Constraints and Integration Challenges.....	54
4.2.4 Intent Recognition Accuracy Implications	54
Chapter 5: Conclusions and Recommendations	55
5.1 Summary and Conclusions	55
5.2 Recommendations for Future Work.....	56
Contribution to Literature	58
References	59
APPENDIX.....	66

APPENDIX A.....	66
APPENDIX B.....	70

List of Tables

Table 1. Comparison of VISA4D with existing methods	9
Table 2. Participant demographics for VISA4D tool validation study	42
Table 3. Sample questions categorized by their perspective commands	45
Table 4. Detailed evaluation of intent	50

List of Figures

Figure 1. Research methodology diagram. (a) Problem identification; (b) Design and development; (c) Implementation; (d) Evaluation.....	16
Figure 2. System workflow and component interaction diagram for VISA4D	21
Figure 3. VISA4D user interface	23
Figure 4. System sequence diagram for input handling.....	24
Figure 5. Navisworks_api.py workflow for task update.....	29
Figure 6. Navisworks schedule after updating the door installation status using VISA4D.....	32
Figure 7. Navisworks schedule after updating the floor, window 100, and window 200 date and status using VISA4D.....	33
Figure 8. Navisworks schedule after adding roof mechanical equipment task, relative dates and status update using VISA4D.....	34
Figure 9. Navisworks schedule after adding date and status update to curtain wall mullion using VISA4D	35
Figure 10. Navisworks schedule after updating using VISA4D.....	36
Figure 11. Navisworks schedule after updating east elevation cladding status to on hold using VISA4D	37
Figure 12. Navisworks schedule after updating the status of west, north and south elevation cladding to in progress using VISA4D	38
Figure 13. Navisworks schedule after updating the status of roof to on hold using VISA4D.....	39
Figure 14. Navisworks schedule after deleting temporary protective canopy installation task using VISA4D	40
Figure 15. Educational building Navisworks schedule after updating using VISA4D	41
Figure 16. Participants' academic standing distribution	43
Figure 17. Participants' tool experience distribution	43
Figure 18. VISA4D accuracy analysis.....	48
Figure 19. Error distribution	49
Figure 20. Confusion matrix	49

Abbreviations

VISA4D	Voice-Integrated Scheduling Assistant for 4D BIM
NLP	Natural Language Processing
BIM	Building Information Modeling
AEC	Architecture, Engineering, and Construction
3D BIM	Three dimensional BIM
4D BIM	Four Dimensional BIM
CAD	Computer Aided Design
AI	Artificial Intelligence
LLMs	Large Language Models
CPM	Critical Path Method
LPS	Last Planner System
PERT	Program Evaluation and Review Technique
ML	Machine Learning
KBES	Knowledge-Based Expert System
NLU	Natural Language Understanding
DL	Deep Learning
GPT	Generative Pretrained Transformer
BERT	Bidirectional Encoder Representations from Transformers
DSR	Design Science Research
API	Application Programming Interface
JSON	JavaScript Object Notation
GUI	Graphical User Interface
APS	Autodesk Platform Services
FEDS	Framework for Evaluation in Design Science
IT	Information Technology

Chapter 1: Introduction and Background

1.1 Background

The architecture, engineering, and construction (AEC) sector is experiencing a significant technological transformation, where traditional methodologies increasingly intersect with advanced computational methods. Building Information Modeling (BIM) has emerged as one of the most cutting-edge and transformative practices in the AEC sector, defined as a combination of technologies, workflows, and guidelines that allows various stakeholders to collaboratively plan, build, and manage a facility within a shared digital environment [1]. The integration of time with three-dimensional (3D) BIM, known as four-dimensional (4D) BIM, has further transformed project delivery by connecting building elements to construction scheduling tasks, enabling visualization and optimization of construction sequences [2] [3] [4].

BIM represents a fundamental paradigm shift in the AEC industry's approach to project delivery. BIM represents a methodology for developing and overseeing digital models that capture both the structural and operational properties of built environments, involving the digital creation of precise virtual replicas of construction projects [1]. Unlike traditional Computer Aided Design (CAD) approaches that focus on geometric representation, BIM employs intelligent, parametric objects that contain detailed information about building components and their relationships within a structured database [1]. This digital repository enables real-time access and sharing of information among project stakeholders, effectively reducing the fragmentation that has historically characterized the construction industry [5]. The transformative nature of BIM extends beyond 3D visualization to encompass a collaborative methodology that integrates all project participants throughout a building's lifecycle [6].

The evolution from 3D to 4D BIM represents a critical progress in construction project management. 4D BIM combines three-dimensional digital building models with time-based scheduling data, enabling project teams to visualize the construction process through animated representations that link building elements to specific construction activities over time [4]. This temporal aspect facilitates spatial-temporal conflict detection, construction sequence optimization, and enhanced communication among project teams [7]. By enabling project teams to simulate construction sequences before actual execution, 4D BIM allows for the identification of potential clashes, resource conflicts, and scheduling inefficiencies that might otherwise remain undetected until construction is underway [8]. The technology has proven particularly valuable in complex projects where traditional scheduling approaches are inadequate, providing capabilities for construction site examination that include temporary element analysis, equipment movement visualization, and safety management through spatial-temporal analysis [9].

1.2 Problem Definition and Research Motivation

Despite these advancements, construction project management, particularly scheduling, remains one of the most challenging aspects of the industry due to its inherent complexity, uncertainty, and the constant need to adjust to changing site conditions [10]. In most cases, developing, tracking, and revising schedules is still largely manual and time-consuming process that lacks integration with on-site activities [11]. The integration of Artificial Intelligence (AI) into construction processes represents a transformative opportunity to address these persistent challenges. Recent advancements in Natural Language Processing (NLP) and the development of Large Language Models (LLMs) have showcased significant improvements in understanding context, processing unstructured information, and facilitating human-computer interaction through intuitive interfaces

[12]. These technological developments offer a unique potential to bridge the operational gap between construction site activities and digital schedule updates and maintenance systems.

Construction scheduling methodologies have evolved substantially, with traditional network-based techniques, like the Critical Path Method (CPM), and Program Evaluation and Review Technique (PERT) advancing alongside lean-based approaches like the Last Planner System (LPS). In parallel, the rise of digital technologies has led to the development of 4D visualization and simulation tools [13], which can be integrated with various planning systems, including LPS, to enhance reliability, communication, and coordination. By integrating time-related data with 3D models, the use of 4D BIM enables project teams to identify potential conflicts early, optimize construction sequences, and improve communication across teams by providing animated representations of the construction process linked to scheduling data [14]. This capability has proven particularly valuable in complex projects where conventional scheduling approaches are inadequate, allowing for thorough testing of construction plans and enhancing communication with project participants [15].

Recent empirical studies have demonstrated the significant potential of voice and natural language interfaces to enhance operational efficiency across various industries [16] [17]. In the context of construction, these technologies offer promising opportunities to transform management practices by enabling more intuitive, accessible, and field-oriented interactions with sophisticated digital systems. An innovative approach combining conversational AI with BIM via cloud infrastructure has emerged based on input from various construction professionals [18]. Site workers using these integrated systems prioritize features like hands-free operation, quick system response, and automatic calculation capabilities when accessing digital building models in field conditions. The implementation of such voice-enabled construction systems offers several

advantages, including better adherence to safety protocols, workforce efficiency gains through instant information access, enhanced project monitoring capabilities, and improved opportunities for on-the-job learning without disrupting workflows [18]. By facilitating direct communication between construction professionals and 4D BIM platforms through natural language, they can reduce documentation burdens, enhance data accuracy, and improve the timeliness of schedule updates.

Several critical problems have been identified in current construction scheduling practices:

1. Administrative burden of project teams and time inefficiency of administrative tasks: Project managers and superintendents typically dedicate between 30-50% of their productive time to documentation and administrative tasks related to progress monitoring and schedule updating [19]. This represents a substantial inefficiency in industry workflows, particularly considering the critical importance of site-based expertise in effective decision-making processes.
2. Technical barriers and accessibility issues of 4D BIM tools: The usual practice with BIM and scheduling software tools predominantly relies on office-based desktop interfaces, creating a functional separation between construction site operations and digital project management systems [20]. This disconnect requires specialized technical expertise and creates barriers for field personnel who may not have advanced training in these tools.
3. Communication gap between field observation and digital documentation: A significant communication gap exists between field observations and the digital documentation system. The requirement for specialized technical expertise and the substantial time investment needed for schedule maintenance and updates create delays in information flow and potential discrepancies between planned and actual progress [21].

4. Limited real-time integration of field operations with digital scheduling systems: Current systems lack effective real-time integration capabilities that would allow immediate updates from field observations to be reflected in digital scheduling systems. This limitation affects the timeliness and accuracy of project information.

1.3 Research Questions

Based on the identified problems, this research addresses the following key research questions:

1. How can NLP and voice interface technologies be integrated with 4D BIM to create an accessible, real-time construction schedule updating and maintenance system that bridges the gap between field operations and digital project management?
2. What system architecture and technological components are required to effectively integrate NLP capabilities with existing 4D BIM platforms like Autodesk Navisworks?
3. How accurate and reliable is NLP-based intent recognition for construction-specific commands, and how can system effectiveness be measured in terms of technical performance and user experience?

1.4 Proposed Approach

Despite significant technological advancements in BIM-based construction scheduling, substantial methodological gaps persist in real-time schedule updating and maintenance. To address the gap in the literature, this research developed a novel framework, named Voice-Integrated Scheduling Assistant for 4-Dimensional Building Information Model (VISA4D), that leverages advanced NLP capabilities within the 4D BIM domain. While existing construction schedule management methods often depend on conventional computer interfaces and office-based data entry, they can limit the immediacy of field-to-digital schedule updates and contribute to higher administrative

workloads. VISA4D establishes more effective connections between field operations and digital scheduling systems. Although specialized knowledge may still be needed to operate such systems effectively, including the current version of VISA4D, this approach represents a step toward streamlining routine updates and reducing inefficiencies in project workflows. Also, it represents a shift in construction schedule management, moving beyond standard computer interfaces to embrace more intuitive and accessible interaction methodologies that accommodate the operational realities of construction environments [22]. While 4D BIM has already demonstrated its value in enhancing safety management, improving risk mitigation strategies, and optimizing construction sequences [23], the integration of natural language interfaces with these systems presents an opportunity to further enhance their accessibility and practical value for construction professionals.

VISA4D is based on the idea that construction schedule management can be significantly improved through natural language interfaces that enable field personnel to interact directly with 4D BIM systems via voice and text commands. This approach acknowledges the irreplaceable value of human expertise in construction management while addressing inefficiencies associated with traditional documentation and schedule updating processes. Therefore, the main contribution of VISA4D is that it addresses the gap in linking site operations with digital project records by integrating natural language interfaces with 4D BIM. This enables field personnel to update schedules via voice or text without directly accessing the BIM model, reducing technical barriers and supporting timely, accurate progress tracking on site.

This thesis presents a comprehensive investigation of VISA4D's conceptual framework, technological implementation, and practical applications within the context of construction scheduling. The research demonstrates how natural language interfaces can transform construction

schedule updating and maintenance by enhancing accessibility, reducing administrative workload, and enabling more effective integration between field observations and digital documentation systems. The findings contribute to the growing body of knowledge on AI-enhanced construction management methodologies and offer practical insights into the design and deployment of natural language interfaces within complex technological environments.

1.5 Contribution of Research

The contributions of the VISA4D framework to the existing body of knowledge are in several critical dimensions as follows:

1. Field-based interaction methods: Unlike conventional BIM interfaces that rely exclusively on office-based interaction, VISA4D establishes direct communication channels between construction site operations and digital scheduling platforms by enabling on-site personnel to interact with the 4D BIM environment through intuitive natural language commands.
2. Multimodal communication protocols: The implementation of both voice and text-based interaction distinguishes VISA4D from existing construction schedule interfaces. This dual-channel approach accommodates diverse operational environments, from construction sites where voice commands allow hands-free operation while performing physical tasks, to office environments where text-based interaction might be preferred.
3. Facilitation of 4D schedule manipulation: The system enables complete schedule management operations through natural language commands in a 4D BIM environment, such as task addition/deletion, modification of time and date parameters, and task status adjustments.
4. Real-time synchronization with BIM: The instant integration of natural language commands with 4D BIM visualizations represents a significant advancement over

traditional methodologies that typically involve substantial processing delays between field observations and schedule updates.

VISA4D directly addresses critical limitations identified in current research regarding AI integration with construction scheduling. Recent studies emphasize the limitations of automated scheduling systems, particularly their reliance on rigid templates and inability to effectively capture human expertise [24]. VISA4D's application of NLU establishes a more intuitive mechanism for translating human judgment into actionable scheduling parameters.

The system also addresses limitations identified in recent evaluations of LLMs in construction contexts. While research has demonstrated potential in generating construction task sequences [25], challenges remain in aligning outputs with project-specific objectives. VISA4D avoids these limitations by focusing on schedule updating and maintenance, keeping decision-making in the hands of project personnel while streamlining the implementation process.

Finally, from a theoretical perspective, VISA4D contributes to the dialogue on human-computer interaction in construction management by demonstrating the potential of natural language interfaces to bridge the gap between field operations and digital systems. This approach aligns with current research emphasizing the importance of developing technologies that accommodate existing workflows rather than imposing artificial constraints. Overall, the development of VISA4D represents a significant advancement in AI integration with construction scheduling, addressing critical gaps in 4D BIM accessibility and efficiency. Table 1 provides a summary of a comparison of VISA4D with existing methods that are elaborated in more detail in the following section, [Chapter 2 Literature Review](#).

Table 1. Comparison of VISA4D with existing methods

Feature	VISA4D	Nabavi et al. (2023) [26]	Singh et al. (2023) [27]	Aljebory and Issam (2019) [28]
Primary Technology	Natural language interface for 4D BIM	NLP for automated information inquiry from BIM	BIM and NLP integration for automatic scheduling	AI-based project planning by connecting BIM with Primavera
Field Usability	Both site and office	Office-based only	Office-based only	Office-based only
Real-time updates to schedule	Real-time synchronization with 4D BIM	N/A	Not focused on real-time operations	Not focused on real-time updates
User Accessibility	Intuitive natural language interaction requiring minimal training	User-friendly but requires some training	User-friendly but limited in generating schedule	Requires expert knowledge to operate
Integration with BIM	Direct integration with 4D BIM	Integration with BIM	Proposes integration but lacks implementation details	No direct integration with BIM
Schedule Manipulation	Complete schedule operations through natural commands	Limited to information retrieval	Focuses on schedule generation rather than manipulation	Limited to schedule generation
Multimodal Communication	Voice and text interaction	Voice and text interaction	Text-based interaction with potential voice interaction	Text-based interaction only

Chapter 2: Literature Review

The integration of AI into BIM represents a significant advancement within the AEC industry. This literature review explores the applications and potential of AI in enhancing 4D BIM processes and construction management.

2.1 Role of AI in Construction Management

AI is transforming construction scheduling by automating complex processes and enhancing decision-making capabilities [29]. In recent decades, numerous studies have explored how AI and its subfields can address industry-specific challenges in construction [30] [31] [32]. One area that faces significant challenges is construction planning, as current systems largely depend on manual input and lack flexibility [24]. AI can significantly advance the automation of construction

scheduling by learning from vast datasets of project records, thus reducing the reliance on manual setup and maintenance of scheduling systems. This advancement could lead to more robust, adaptable systems that learn and improve from each executed project, making the scheduling process more efficient and less prone to human error [24].

Several AI applications have been successfully integrated into construction scheduling with promising results. In [33], EUROPA provides a comprehensive platform for constraint-based temporal planning that has been successfully implemented in NASA projects. This framework offers capabilities in three key areas: representation, reasoning, and search. EUROPA's modular architecture allows extensive customization through plugins and extensions, making it adaptable to various scheduling challenges, though it requires significant technical expertise to implement [33].

Building on this foundation, Knowledge-Based Expert System (KBES) has been developed in [28] that connects Autodesk Revit with Primavera. Their system automates the extraction of building components from BIM models and converts them into scheduled activities with proper sequencing. Unlike EUROPA, which requires specialized knowledge [33], this system creates a more accessible bridge between design and scheduling by leveraging existing commercial software familiar to industry professionals. The authors demonstrated through a simple house building case study that their approach eliminates many tedious steps in schedule creation, though field testing on complex projects remains limited [28].

A critical review of automated planning methods spanning three decades were conducted in [24], identifying key barriers to adoption: (1) rigid knowledge representation in existing systems; (2) dependency on manually maintained work templates; (3) lack of automated learning methods; (4) limited validation on real-world projects; and (5) disconnection between planning

and optimization research. Their analysis suggests that advancements in deep learning (DL) and NLP could facilitate automatic generation of dynamic work templates from existing project records, reducing the manual effort currently required.

While these approaches represent significant advancements in integrating AI with construction scheduling, they still face implementation challenges. The trend toward more intuitive interfaces is evidenced in recent research combining Natural Language Understanding (NLU) with BIM [34], facilitating voice-activated updates to improve accessibility for field personnel. The integration of ML-enhanced BIM models with natural language capabilities presents opportunities to improve schedule management efficiency and reduce technical barriers to adoption [35] [36] [34].

Despite these advancements in AI applications for construction scheduling, a significant gap remains in real-time, field-accessible interfaces that allow project managers to interact naturally with scheduling systems. While [24] highlighted the rigid knowledge representation and limited real-world validation of existing systems, and both [33] and [28] demonstrated powerful AI planning capabilities; these solutions still require specialized technical knowledge and office-based operation. The VISA4D framework addresses this critical gap by leveraging AI to process natural language commands delivered through voice and text inputs directly from construction sites. Unlike the existing systems described above, VISA4D democratizes access to scheduling technology by enabling immediate on-site schedule updates through intuitive voice commands, thereby reducing the technical barriers that have historically limited the adoption of advanced scheduling tools among field personnel.

2.2 Natural Language Processing in BIM and Construction Scheduling

In recent years, NLP has found significant applications within BIM information systems, effectively reducing the need for extensive knowledge and technical expertise among users. Users can access and manipulate BIM data through natural language queries, facilitating interaction without requiring an in-depth understanding of the software's functionalities [37]. Advanced LLMs, such as OpenAI's Generative Pre-trained Transformer (GPT) series and Bidirectional Encoder Representations from Transformers (BERT), have demonstrated remarkable efficiency in tasks involving comprehension, generation, and reasoning with natural language by leveraging DL techniques and large-scale datasets to develop relative representations of text [38]. This capability allows them to generate consistent and contextually appropriate responses, thereby enhancing user engagement with BIM platforms.

Leveraging existing technological frameworks, researchers have advanced several innovative NLP-enhanced BIM systems. For example, a BIM-GPT framework was introduced in [37] which effectively translates natural language queries into relevant segments of 3D BIM models. Similarly, a GPT-powered assistant named DAVE was designed for real-time multimodal interactions within BIM contexts [39]. This system integrates with Autodesk Revit, enabling users to engage through both text and voice for updates and inquiries, thus optimizing complex BIM workflows. Additionally, a chatbot named JULIE was tailored specifically for construction site managers. JULIE employs NLU to identify user intent accurately, facilitating the delivery of actionable insights via conversational interfaces [34].

NLP has shown considerable promise in optimizing planning and scheduling processes within the construction sector. A comprehensive evaluation of the current landscape of automated planning and scheduling in construction is provided in [24]. The investigation primarily focused on identifying the main challenges that have caused the delay in the adoption of automated planning techniques and systems within the construction industry. A notable finding is the difficulty in transferring the expertise of human planners to automated systems, mainly due to the absence of standardized terminology and systematic knowledge frameworks in the construction domain. The authors advocate for the application of NLP to bridge the gap between short-term planning tasks, such as look-ahead scheduling, and long-term planning tasks, such as master schedules within the construction industry [24]. Furthermore, the study critiques existing methodologies for their heavy reliance on rigid manual templates and highlights the critical need for adaptable knowledge management systems. These insights underline the imperative to refine current practices by integrating flexible, automated solutions that can effectively leverage human expertise in construction planning [24].

In a recent study, researchers investigated the influence of GPT models on project management, specifically concerning scheduling and task allocation in construction projects [25]. The primary objective was to provide a comprehensive analysis of tasks generated by these models. The results demonstrated that GPT models could create a coherent and sequential task list across various scenarios. However, it was noted that some generated tasks did not fully align with the project's overarching goals. The quantitative analysis revealed variability in the project timeline for different stakeholders relative to the baseline schedule. Furthermore, discrepancies were observed in the number of personnel required to complete the proposed tasks [25].

Despite these advances, current NLP implementations in construction scheduling still have limitations in real-time on-site application and voice command integration. The proposed VISA4D framework addresses this gap by enabling direct voice and text command inputs from construction sites to instantly update 4D BIM models, potentially transforming how construction schedules are managed and communicated across project teams.

2.3 Real-Time Schedule Updating

In construction management, real-time schedule updating represents a critical yet challenging aspect of project oversight. This process involves comparing actual construction data from the site with the initial plan to accurately adjust activity timelines. The scheduling of construction activities relies mostly on the expertise of superintendents and project managers, who conduct visual inspections and documentation based on their direct oversight and knowledge [40].

A common application of BIM in construction involves linking schedules to BIM models, creating 4D BIM, a combination of 3D models with time information. This integration enables visualization of construction simulations and provides essential insights into critical schedule aspects [11]. The process necessitates a thorough assessment of on-site project conditions and a comparison between current progress and baseline schedules [11] [41].

To support this complex process, researchers have proposed various technological solutions. These include frameworks incorporating automated activity tracking, material tracking, and mobile computing communication [42], as well as methods integrating 4D BIM with 3D data for construction progress measurement [43]. A two-stage revision process was developed to address incomplete data sets and it achieved progress tracking accuracy of up to 100% [43]. Building on these innovations, a system was created to streamline schedule updates by fusing 3D

point-cloud data with 4D BIM, enhancing accuracy through automated registration techniques and effective object matching [19].

Despite advancements in technology, schedule updating and maintenance in construction still rely heavily on human expertise. The VISA4D framework acknowledges the indispensable role of human judgment within construction management. Rather than aiming for full automation, it focuses on streamlining the labour-intensive tasks of data collection and organization involved in progress monitoring, and therefore, enhancing documentation efficiency while preserving the value of expert insight.

Chapter 3: Research Methodology and Implementation

3.1 Research Methodology

This research employs a design science research (DSR) methodology [44] to develop and evaluate the VISA4D framework. DSR is particularly suitable for this study as it focuses on creating innovative artifacts to solve practical problems while making rigorous theoretical contributions to knowledge [44]. DSR is characterized by its emphasis on developing solutions to address problems in a domain [45], which aligns perfectly with the aim of improving construction schedule management through natural language interfaces to 4D BIM.

Following the DSR methodology guidelines established in [44] and extended in [46], this research was structured into four main phases: (1) problem identification and conceptualization, (2) design and development, (3) implementation, and (4) evaluation, as illustrated in Figure 1.

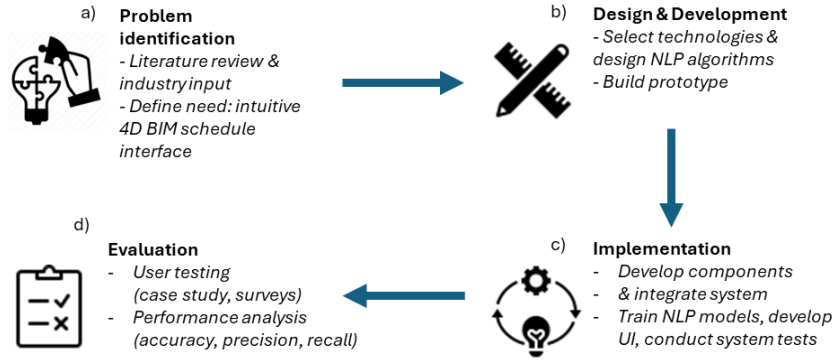


Figure 1. Research methodology diagram. (a) Problem identification; (b) Design and development; (c) Implementation; (d) Evaluation

3.1.1 Problem Identification

The initial phase involved identifying the research problem through an extensive literature review, professional experience, and insights gathered from project management and site supervision staff (Figure 1.a). This approach aligns with the principle that DSR projects should begin with a problem relevant to the application environment [47].

This phase established the need for improved methods to update and maintain construction schedules in 4D BIM environments. The literature review revealed that while 4D BIM offers significant advantages for construction scheduling, the updating process remains complicated and slow, particularly for field personnel who may lack advanced training in specialized BIM tools. Furthermore, input from construction professionals highlighted the communication gap between field operations and office-based BIM specialists, leading to delays in schedule updates and potential discrepancies between planned and actual progress.

The problem identification phase was guided by principles from DSR [48], incorporating both theoretical gaps from the literature and practical challenges raised by industry professionals, to support relevance and potential contribution. Based on these findings, the research problem was conceptualized as a need for an intuitive interface that allows direct interaction with 4D BIM

schedules through natural language commands. This conceptualization guided the development of system requirements and the overall architecture of VISA4D.

3.1.2 Design and Development

The design and development phase (Figure 1.b) focused on creating the system architecture and determining the appropriate technologies and methodologies for implementation. This phase embodies what is described as the creative process of generating new artifacts to address identified problems [49].

The design process involved requirements analysis, system architecture design, technology selection, and algorithm design. Requirements analysis defined functional and non-functional requirements based on the identified problem and user needs. Key requirements included the ability to process natural language commands, integration with a commonly used 4D BIM software, such as Autodesk Navisworks, support for both voice and text input, and real-time schedule updates.

System architecture design ([Chapter 3, Section 3.2](#)) developed a modular system where each major function, the user interface, NLP processing, task management, and BIM integration, was designed as a separate, dedicated component with its specific role. This approach follows the modularity principle adopted in DSR by [50], making the system more flexible and easier to maintain.

Technology selection involved choosing appropriate technologies for each system component, including Python for client-side components due to its extensive NLP libraries, C# for server-side integration with Navisworks, and JavaScript Object Notation (JSON) for data interchange. Algorithm design included designing the NLP pipeline and algorithms for intent

classification, entity extraction, and command validation. This involved developing machine learning (ML) models and rule-based patterns for construction-specific language processing.

The development process incorporated iterative refinement, which is a key characteristic of DSR as described in [51]. The system was implemented using modern software development practices, including object-oriented programming and asynchronous processing. Regular prototype testing was conducted throughout this phase to identify and address technical issues, as recommended for continuous evaluation in DSR projects [52].

3.1.3 Implementation

The implementation phase (Figure 1.c) translated the design specifications into a functional system, embodying the emphasis on creating tangible artifacts as the primary contribution of DSR [44]. This phase involved several key activities, including component development, integration, NLP model training, user interface implementation, and system testing.

Component development implemented each system component according to the architectural design of the system, including the Graphical User Interface (GUI) layer, NLP processor, task manager, Navisworks Application Programming Interface (API), and custom Navisworks plugin. Through well-defined interfaces and protocols, as explained in more detail in Section 4, communication has been established between all components. This included implementing the HTTP server and RESTful API for interaction between the client application and Navisworks.

NLP model training involved training and fine-tuning the intent classification model using construction-specific training data generated from templated command structures. User interface implementation developed an intuitive interface that supports both text and voice inputs, with appropriate feedback mechanisms and error handling. System testing included conducting iterative

testing to ensure functionality across various command types and scenarios. The testing strategy encompassed unit testing to validate individual components individually, followed by integration testing to assess the functionality of the complete system as an integrated system.

The implementation resulted in a functional prototype of the VISA4D framework capable of processing natural language commands and executing corresponding actions within the Navisworks environment.

3.1.4 Evaluation

The evaluation phase (Figure 1.d) assessed the effectiveness, usability, and performance of the VISA4D framework through multiple approaches. This comprehensive evaluation strategy follows the Framework for Evaluation in Design Science (FEDS) [46], which emphasizes the importance of a multi-faceted evaluation approach for DSR artifacts.

As described in more detail in [Chapter 3, Section 3.4](#), user testing was conducted with 20 undergraduate and graduate Civil Engineering students who used the VISA4D tool for the case study of a real-world office building construction. Participants performed various tasks, including adding new tasks, deleting existing tasks, updating dates, and modifying task statuses using both voice and text commands. Qualitative and quantitative feedback were collected through anonymous surveys that included structured multiple-choice questions and open-ended questions to evaluate user experience, ease of use, and observed utility. This approach aligns with guidance on user-centered evaluation of information technology (IT) artifacts [53].

Performance testing ([Chapter 3, Section 3.4](#)) involved developing a systematic testing framework to assess the reliability and accuracy of the NLP processor in interpreting construction-specific commands. This included testing 80 unique commands across five intent categories and analyzing the results using various performance metrics. Accuracy analysis computed and

analyzed metrics such as intent accuracy, task name extraction accuracy, precision, recall, and F1-scores to identify strengths and weaknesses in the system's NLP capabilities.

Finally, the evaluation phase provided comprehensive insights into the VISA4D tool's performance and usability, validating its effectiveness while also highlighting areas for future improvement. This combination of naturalistic (i.e., user testing) and artificial (i.e., performance testing) evaluation strategies follow the balanced approach recommended in [46] for robust DSR evaluation.

3.2 System Architecture and Implementation Details

The implementation of the VISA4D tool follows a modular approach, with each component designed to fulfill specific functions within a voice and text-interactive framework. This section details the system architecture and code structure of the VISA4D tool that integrates conversational interfaces with Autodesk Navisworks. All source code for the VISA4D tool was custom developed and is available in an open-source repository [54].

Modern software development practices, including object-oriented programming and asynchronous processing, have been utilized in this process to ensure a responsive user experience. Python programming language was used for the development of client-side components as it offers extensive libraries and frameworks for NLP to perform text preprocessing, sentiment analysis, and named entity recognition. Additionally, Python offers various tools and widgets through its user interface libraries, such as *Tkinter*, which is ideal for building *chat-bot-type* interfaces.

On the server side, integration with Navisworks is achieved using C#, which directly interfaces with the Autodesk API stack through Microsoft Visual Studio. The codebase is organized into logical modules to ensure a clear separation of interfaces between different components.

As shown in Figure 2, the system comprises many interconnected components, each with specific functions, however, there are three primary components: The NLP processor component (Figure 2.a) is the NLU engine, implemented in Python to leverage its extensive libraries for language understanding, pattern matching, and intent classification. The other two are *NavisworksServer* (Figure 2.f) and *VISA4D Plugin (custom Navisworks plugin)* (Figure 2.g), which are closely integrated components implemented in C# to interface directly with Autodesk Navisworks.

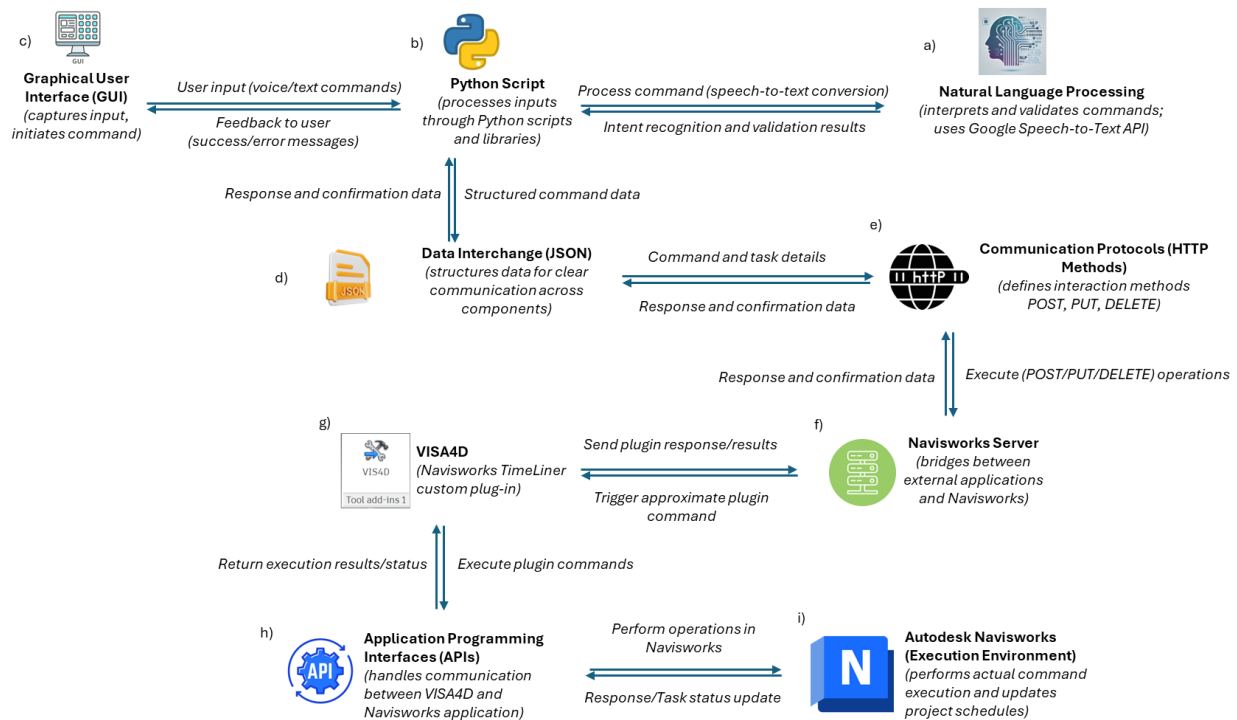


Figure 2. System workflow and component interaction diagram for VISA4D

NavisworksServer component (Figure 2.f) implements a local HTTP server using the *.NET System.Net.HttpListener* class, creating a bridge between external applications, such as the user interface, and Navisworks (Figure 2.i). Whereas the VISA4D Plugin (Figure 2.g) component implements the custom Navisworks plugin, which integrates directly with Navisworks through the Autodesk API.

JSON serves as the primary data interchange format throughout the system (Figure 2.d). It facilitates structured communication across system layers for transmitting parsed commands. HTTP serves as the primary communication protocol between system components (Figure 2.e), implementing RESTful design principles and using HTTP methods, such as POST for creating new tasks in Navisworks *Timeliner*, PUT for updating existing tasks, DELETE for removing tasks from the system, and GET for retrieving authentication status. The system implements *OAuth 2.0* (Open Authorization) authentication for Autodesk Platform Services (APS), ensuring secure access to construction data. In the following section, each of the components is explained in more detail.

3.2.1 User Input layer (gui.py)

The GUI layer (Figure 2.c) serves as the primary interface between users and the VISA4D tool, and facilitates both text and voice-based interactions. This layer was implemented using the *customtkinter* library, which extends Python's *Tkinter* capabilities to create a more refined and interactive interface. The dual-input modality is a key implementation feature of VISA4D, which allows users to interact through both conventional text inputs and voice commands (Figure 3). The voice input functionality, particularly valuable for enabling hands-free operation for site superintendents, was implemented using *SpeechRecognition* library, utilizing Google's Speech-to-Text API as its transcription backend.

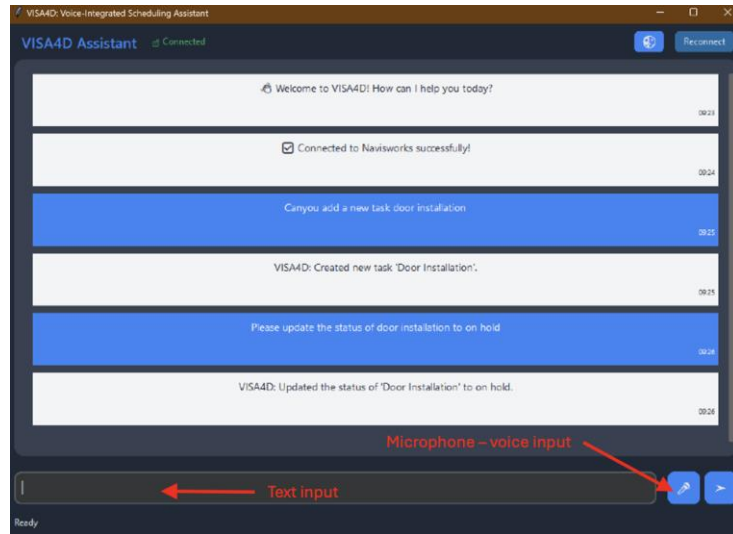


Figure 3. VISA4D user interface

Upon activation through the microphone button (as shown in the system sequence diagram in Figure 4), the system initiates a sequence of operations that begins with microphone calibration using *SpeechRecognition*, followed by audio recording with appropriate timeout mechanisms. The recorded audio undergoes processing and conversion to text, and is treated as a text command moving forward, which is routed to the NLP system for analysis and task execution. If the input is a text command, then the processing is the same except for the microphone calibration.

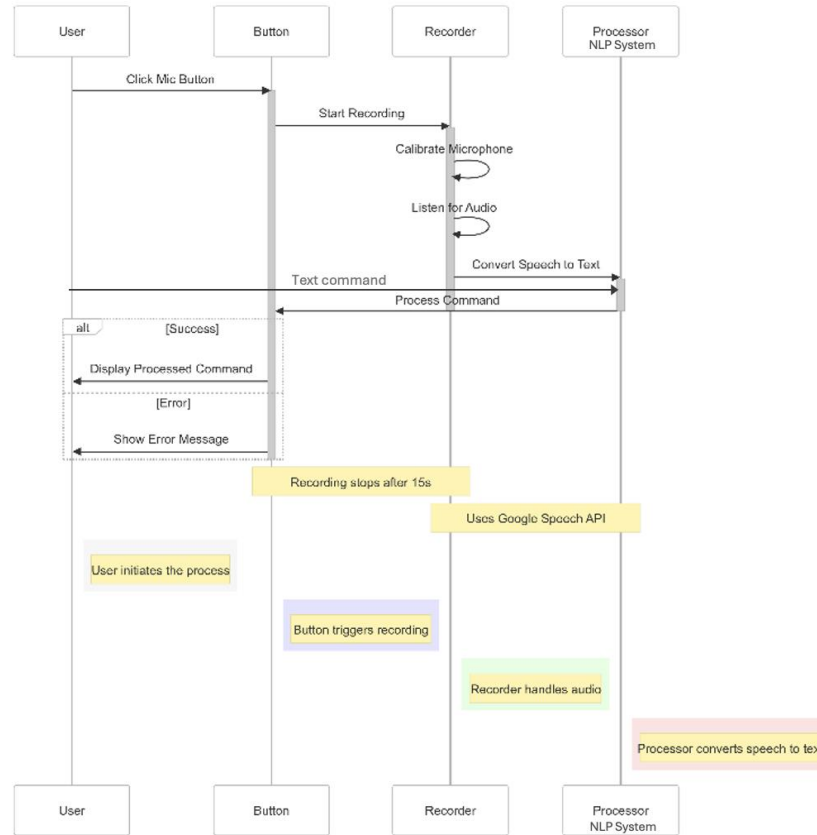


Figure 4. System sequence diagram for input handling

Asynchronous Speech Processing was applied, and speech recognition operations are run in separate threads to prevent user interface blocking or freezing during audio processing operations. Thread synchronization was managed using *Tkinter's* *after* method to safely update user interface elements from worker threads without causing race conditions. Additionally, the implementation incorporates on-demand component initialization to reduce application startup time.

Comprehensive error handling was integrated throughout the GUI layer (Figure 2.c) to manage possible issues such as connection failure or unrecognized speech patterns. To enhance usability across different environments and user preferences, the interface implements a dynamic theming system supporting both light and dark visual modes.

The *OAuth 2.0* authentication mechanism was implemented to secure integration with the Navisworks API. Once authenticated, the status indicator in the header changes to reflect the

connection state (e.g. Connected, as in Figure 3), providing continuous visual feedback about system connectivity. Moreover, the interface actively checks authentication status before attempting protected operations, ensuring system security and preventing unauthorized access.

The GUI layer connects directly to the NLP processor class and task manager class components to forward natural language inputs (both typed and transcribed) to appropriate subsystems for transformation into structured commands. Overall, the GUI layer achieves a balance between visual appeal and functional robustness. Creating a system capable of reliable operation in real-world construction environments.

3.2.2 nlp_processor.py

The *nlp_processor.py* module (Figure 2a) integrates state-of-the-art NLP tools with domain-specific knowledge to interpret and process construction-related commands. Its primary objective is to convert unstructured natural language input into structured, actionable tasks suitable for construction management systems. This module combines multiple techniques such as ML, rule-based pattern matching, and semantic similarity measures to ensure precise and reliable interpretation of user inputs in a construction context.

The NLP pipeline begins with text preprocessing, where raw user input is normalized. Stop words are removed and lemmatization is applied to reduce noise. For example, in the command “Update the foundation pouring task to complete”, tokenization and part-of-speech tagging are handled using the *spaCy* library, which identifies the syntactic roles of words such as “foundation” (noun) and “pouring” (verb). [Appendix A](#), Figure A1 shows an example of NLP processing pipeline.

The intent classification phase is designed to accurately categorize user commands based on their specific operational intent. This includes commands for creating tasks, updating statuses

or dates, and deleting tasks. The classification system encompasses four primary intent categories: `create_task`, `update_status`, `update_date`, and `delete_task`. Any command that does not conform to these defined categories is classified as unknown. [Appendix A](#), Figure A2 shows an example of creating a task.

To support intent recognition, the system uses *TF-IDF vectorization* combined with a *RandomForestClassifier*. The TF-IDF model emphasizes construction-specific terms such as “pour”, “foundation”, and “inspection”, enabling the classifier to distinguish between domain-relevant variations. Training data for this classifier are generated programmatically using templated command structures to ensure coverage of key variations in phrasing.

Once the intent is identified, the system performs entity extraction to delineate task names, statuses, and relevant dates. [Appendix A](#), Figure A3 shows an example of updating date and Figure A 4 shows an example of modifying the status.

Entity recognition is accomplished through a hybrid approach that primarily relies on direct pattern matching with domain-specific templates. If patterns fail to match, it falls back on dependency parsing and noun phrase extraction, leveraging *spaCy*’s syntactic analysis. In cases of ambiguous expressions, semantic similarity is evaluated using *SentenceTransformer*, a BERT-based model that maps text into vector space for comparison. This allows the system to interpret commands like “start concrete pouring” and “begin concrete work” as semantically similar.

The *nlp_processor.py* also incorporates a command validation phase to ensure completeness before execution. It assesses the presence of essential parameters and analyzes the confidence scores associated with identified intents and entities. When the confidence score falls below a predetermined threshold (typically 60%), the system prompts the user for clarification; if between 60–85%, then the system performs additional validation checks before executing the

command. For instance, if it is an “update status” command, the system ensures it has status information. Furthermore, if the confidence score exceeds 85%, the processor proceeds directly to execute the task. This decision framework is detailed in Figure 4, which maps out the end-to-end flow from raw input to task manager execution.

3.2.3 Task_manager.py

The *task_manager.py* (Figure 2d) module serves as a bridge between the NLP processor component and the Navisworks application. Upon interpreting user commands and extracting intents and entities, *nlp_processor.py* forwards this information to *task_manager.py*. This module processes the data and transforms these into a structured JSON format that aligns with the requirements of the Navisworks API.

The module implements four fundamental operations:

1. Task creation: this operation facilitates adding new tasks to the TimeLiner, the 4D scheduling tool in Navisworks.
2. Status updates: this function allows for the modification of task statuses, enabling dynamic transitions between states such as “not started” and “in progress”.
3. Date modifications: this capability provides the flexibility to reschedule tasks by modifying their start or end dates, ensuring accurate project timelines.
4. Task deletion: this operation permits the removal of tasks from the system, effectively managing the project scope and maintaining data integrity.

Communication with Navisworks is handled via a RESTful API using the Python requests library for executing HTTP operations. This interaction is secured using *OAuth 2.0* for authentication and employs JSON formatting for payloads to facilitate data exchange. Additionally, URL encoding is

utilized to handle special characters in task names, along with comprehensive validation and error-handling mechanisms.

A synchronization pattern is implemented where updates are initially applied to a local state before being pushed to Navisworks. The local state is sustained through two JSON-based data structures: (1) `task_statuses` and (2) `task_dates`. This local storing mechanism minimizes the frequency of API calls and enhances overall response times.

Finally, *task_manager.py* also implements comprehensive logging throughout the entire process for effective monitoring and troubleshooting. Logs are stored locally in a *visa4d.log* file where information about operation attempts with parameters, successful completions, error conditions with details, and changes in authentication status are stored. These logs provide visibility into system operations and serve as an audit trail for construction schedule modifications.

3.2.4 Navisworks_api.py

The *navisworks_api.py* module (Figure 2h) facilitates direct interaction between the underlying system and Autodesk Navisworks. It acts as an intermediary that converts user-driven commands into executable tasks. Upon initialization, the module sets up a base URL (<http://localhost:5000>) for interfacing with the Navisworks service. It also defines the requisite standard HTTP headers to facilitate JSON communication and configures endpoint mappings for essential operations, including task creation, updates, deletions, and user authentication. The authentication workflow initiates with the `authenticate` method that retrieves access tokens via client credentials. These tokens, along with their expiration timestamps, are stored locally and securely to ensure their integrity. To maintain token validity, expiration is actively tracked with time buffers. This class implements three fundamental task operations, as discussed previously: `create_task`, `update task`, and `delete task`. The *navisworks_api.py* leverages the Python requests library for HTTP operations,

using appropriate request types for different operations. Each request is constructed with appropriate headers, JSON payloads, and error handling to ensure reliable communication with the Navisworks system. Figure 5 shows the *navisworks_api.py* steps in detail.

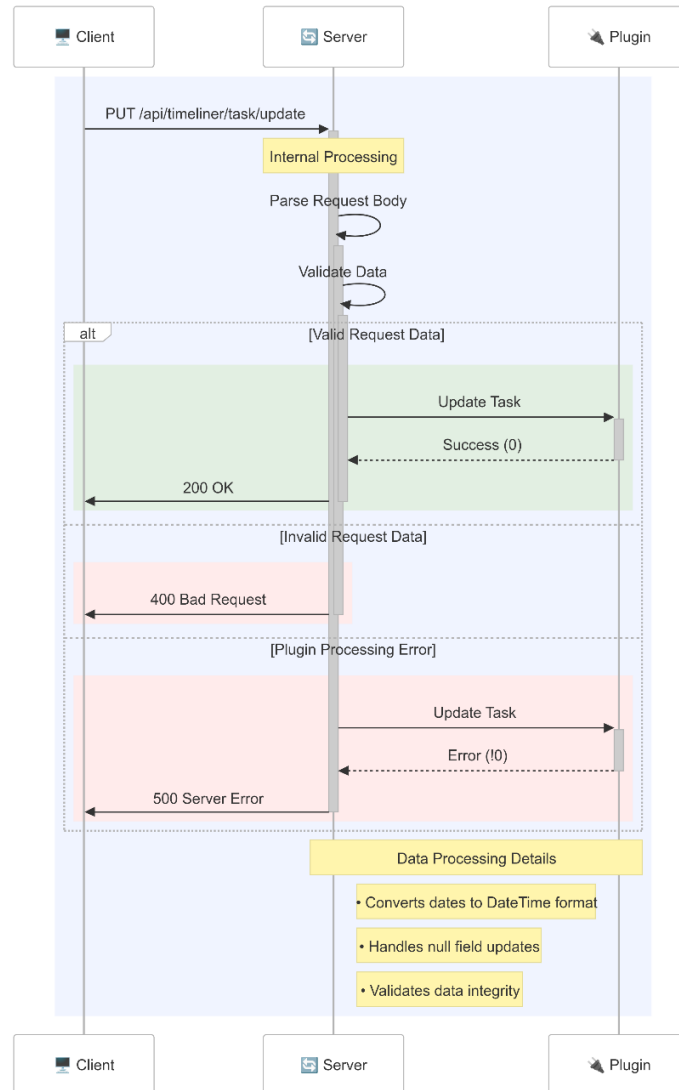


Figure 5. *Navisworks_api.py* workflow for task update

3.2.5 NavisworksServer.cs

The *NavisworksServer.cs* class (Figure 2f) serves as a local HTTP server that facilitates communication between external applications and the custom Navisworks plugin. Operating on

localhost at port 5000, it leverages the `HttpListener` class from the *System.Net* namespace. This setup creates a bridge that allows external applications to interact with Navisworks' TimeLiner functionality through standard web protocols.

The server implementation employs asynchronous programming patterns using C#'s *async/await* functionality, ensuring efficient handling of multiple requests without blocking the main application thread.

The TimeLiner task management system exposes three core REST API endpoints for handling tasks: creation, updating, and deletion. These endpoints are designed to accept JSON payloads for initiating new tasks, processing update requests, and triggering task deletion. The server utilizes *Newtonsoft.Json* for serialization and deserialization, ensuring reliable data exchange between the client and server. [Appendix A](#), Figure A5 shows the three core REST API endpoints for handling tasks.

The server maintains a reference to the main custom VISA4D plugin instance, allowing it to delegate task operations to the appropriate plugin methods. This integration ensures that all operations performed through the API directly affect the Navisworks environment.

3.2.6 Navisworks TimeLiner Custom Plugin (Addin.cs)

The VISA4D Plugin (Figure 2g), the custom Navisworks TimeLiner Plugin, represents the final component of the integration between the NLP system and Autodesk Navisworks. This custom plugin was implemented in C# using Visual Studio 2022. This plugin serves as the interface between the NLP-driven command layer and the Navisworks environment, allowing natural language inputs to trigger direct actions within the 4D environment. The plugin is designed as an *AddInPlugin*, utilizing three fundamental components of the Autodesk API: `Autodesk.Navisworks.Api` for document access and manipulation,

Autodesk.Navisworks.Api.Plugins for interface integration, and Autodesk.Navisworks.Api.TimeLiner for scheduling capabilities.

A key feature of the plugin is its initialization of a local HTTP server (*NavisworksServer.cs*), which listens for incoming commands from the external client application. This client, implemented in Python, handles natural language input and connects to the plugin via the HTTP interface. This architecture enables full-cycle command execution, from language input to 4D model manipulation, supporting task creation, updates, and deletions within Navisworks. Visual feedback is provided through a colour-coding scheme that represents task statuses directly in the 3D model. The plugin uses the colour assignment to represent different task states: gray for “not started”, blue for “in progress”, red for on hold, and green for completed. These colour indications allow users to visually assess construction progress at a glance by inspecting the associated model elements. To ensure data consistency, the plugin’s update mechanism duplicates the original task before any changes are made. This approach protects nontargeted task attributes, ensuring that only the specified parameters are modified. Figure A6 in [Appendix A](#) shows the overall communication data flow for VISA4D.

3.3 Use Case Scenarios

3.3.1 Office Building Construction

During a routine inspection of an office building that is under construction, the site superintendent enters the site to monitor the progress of ongoing tasks. While walking through the site, he identifies a critical issue: the door installation, scheduled to start on January 9, 2025, was suspended due to an unresolved design conflict with the interior partitions. Recognizing the impact, he issues a voice command: “Hey VISA4D, set all door installations on hold”. Instantly,

all door elements in the BIM model turn red, to indicate on-hold status. This real-time visual update triggers immediate coordination among on-site personnel and remote team members, leading to a rapid resolution meeting with the design consultants (Figure 6).

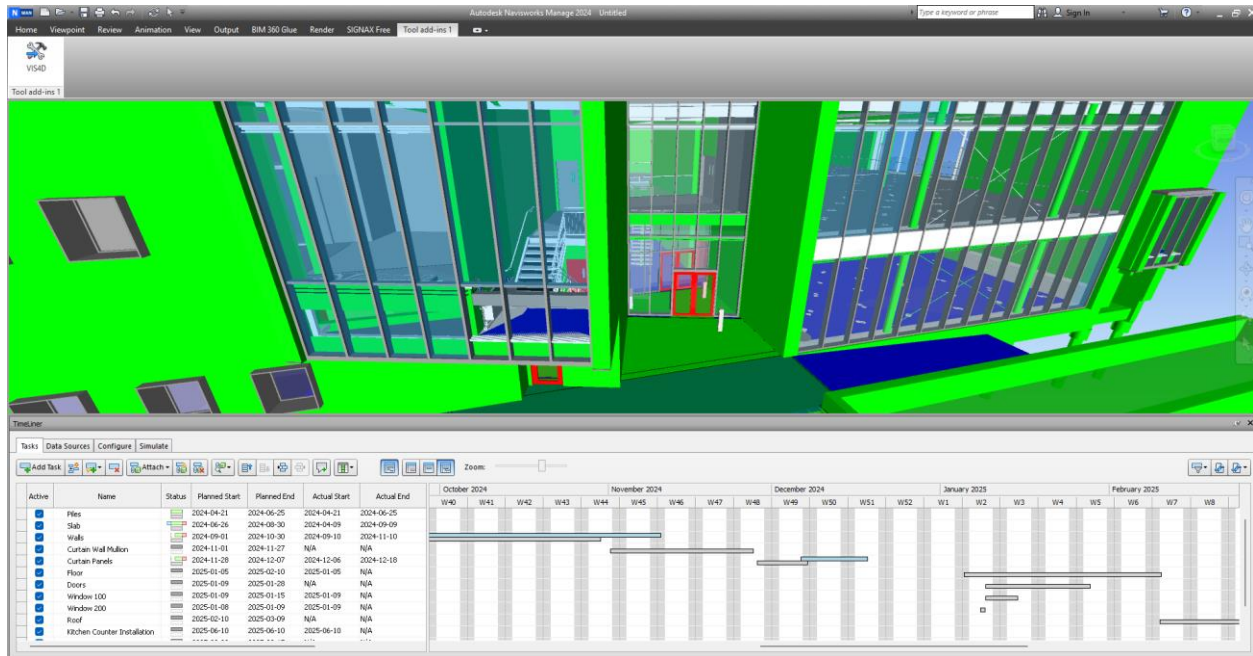


Figure 6. Navisworks schedule after updating the door installation status using VISA4D

Continuing the inspection, the superintendent notices that installation had begun for the floor finishes, Window 100, and Window 200, but these tasks were not yet reflected in the schedule. Using VISA4D’s voice input feature, he states: “Update actual start date for Floor to January 5th, Window 100, and Window 200 to January 9th”. The system responds by adding the dates and shifting the status of those elements from “Not Started” (grey) to “In Progress” (blue), ensuring alignment between physical progress and the digital model (Figure 7).

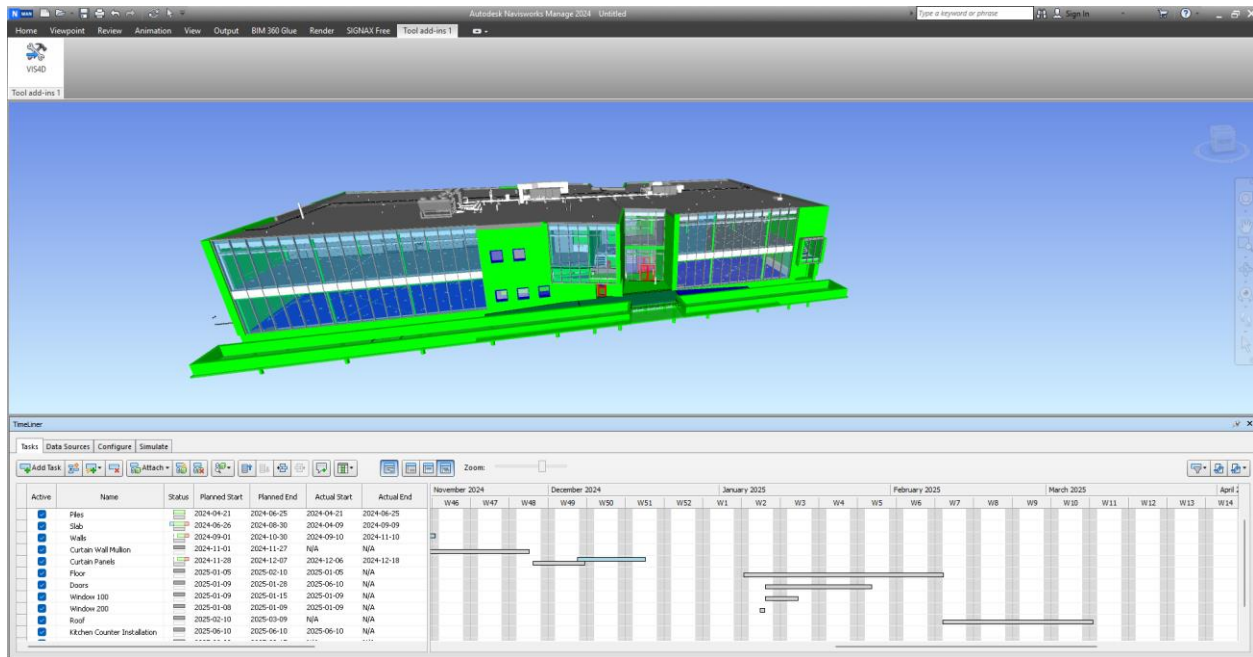


Figure 7. Navisworks schedule after updating the floor; window 100, and window 200 date and status using VISA4D

Upon reaching the roof area, the superintendent remembers a notification received earlier that morning from the mechanical equipment supplier about a two-week delivery delay. Attempting to update the task, he says: “Update the status of roof mechanical equipment installation to on hold”. VISA4D replies that the task did not yet exist. He then instructs: “Add roof mechanical equipment task. Set start date to March 9 and finish date to March 15”. The system adds the new activity, applies the specified dates, marks the task as on hold, updates the Gantt chart and turns the component red in the model to reflect the status (Figure 8). The superintendent finishes his walkthrough and updates all the building elements similarly using voice and text commands on his mobile device.

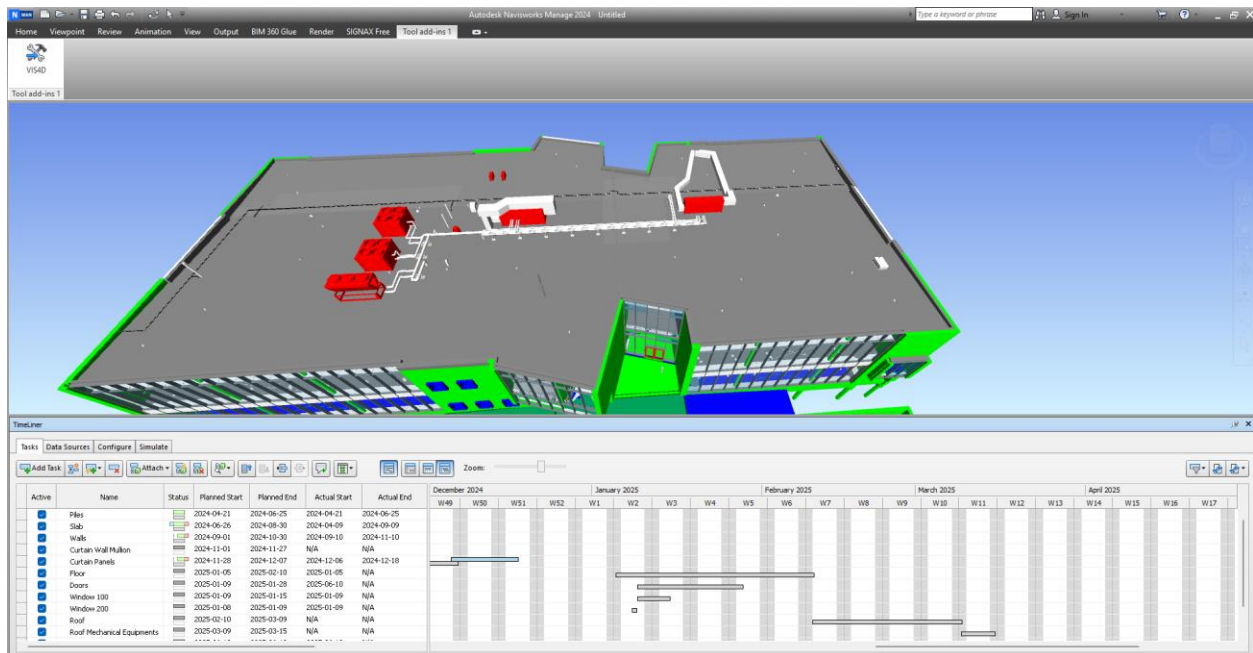


Figure 8. Navisworks schedule after adding roof mechanical equipment task, relative dates and status update using VISA4D

Later that day, during the contractor team’s progress meeting, the project manager reviews the revised schedule in Navisworks. He notices that curtain wall mullions still appear grey, indicating they were not started. The team clarifies that installation had been completed before the Christmas break. The project manager then instructs: “Update the actual finish date for all Curtain wall mullions to December 5, 2024”. The model gets updated accordingly, turning the curtain wall mullions green to indicate completion and adjusts the project timeline (Figure 9).

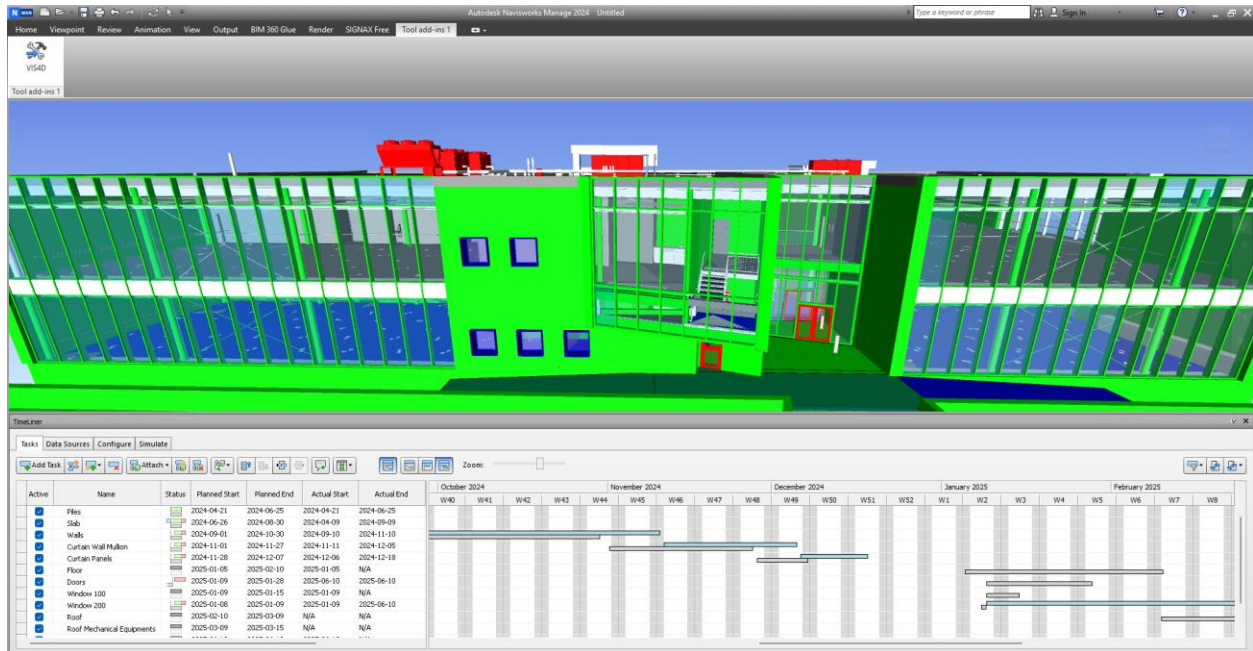


Figure 9. Navisworks schedule after adding date and status update to curtain wall mullion using VISA4D

The updated schedule in Navisworks presents a colour-coded representation of the project status: completed tasks appear in green, on hold tasks are in red, not-started tasks are in grey and in-progress tasks are in blue. This intuitive visual feedback, and VISA4D's seamless integration of voice and text control enables the project manager to quickly assess the project's current state, flag critical issues and identify priority areas for the team's attention, significantly enhancing decision-making efficiency in project management. Figure 10 displays the VISA4D interface, a 3D model with live status indicators integrated with a Gantt chart view and shows how the Navisworks schedule and the Gantt chart are updated by VISA4D.

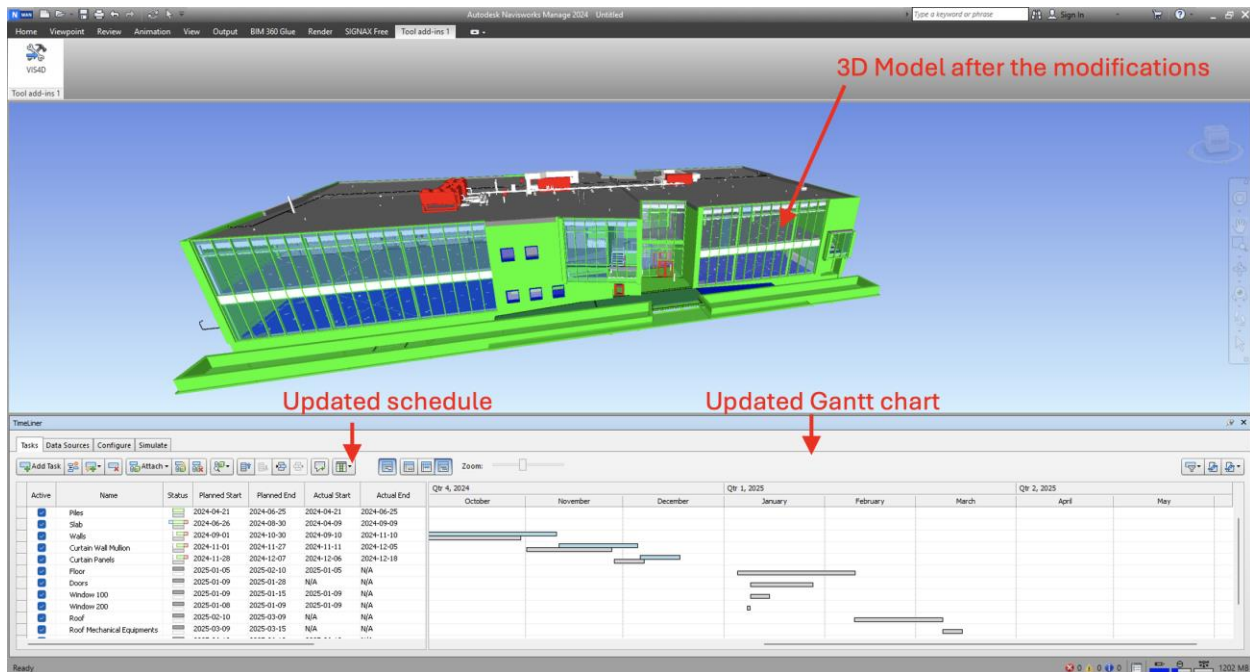


Figure 10. Navisworks schedule after updating using VISA4D

3.3.2 Educational Building Construction

The project superintendent receives an email from the supplier notifying her of a delay in the delivery of cladding materials needed for the east elevation. Right from her desk, the superintendent opens the VISA4D interface on her desktop and uses a text command: “Update east elevation cladding status to on hold”. The schedule updates instantly and the change gets reflected in the 4D model (Figure 11).

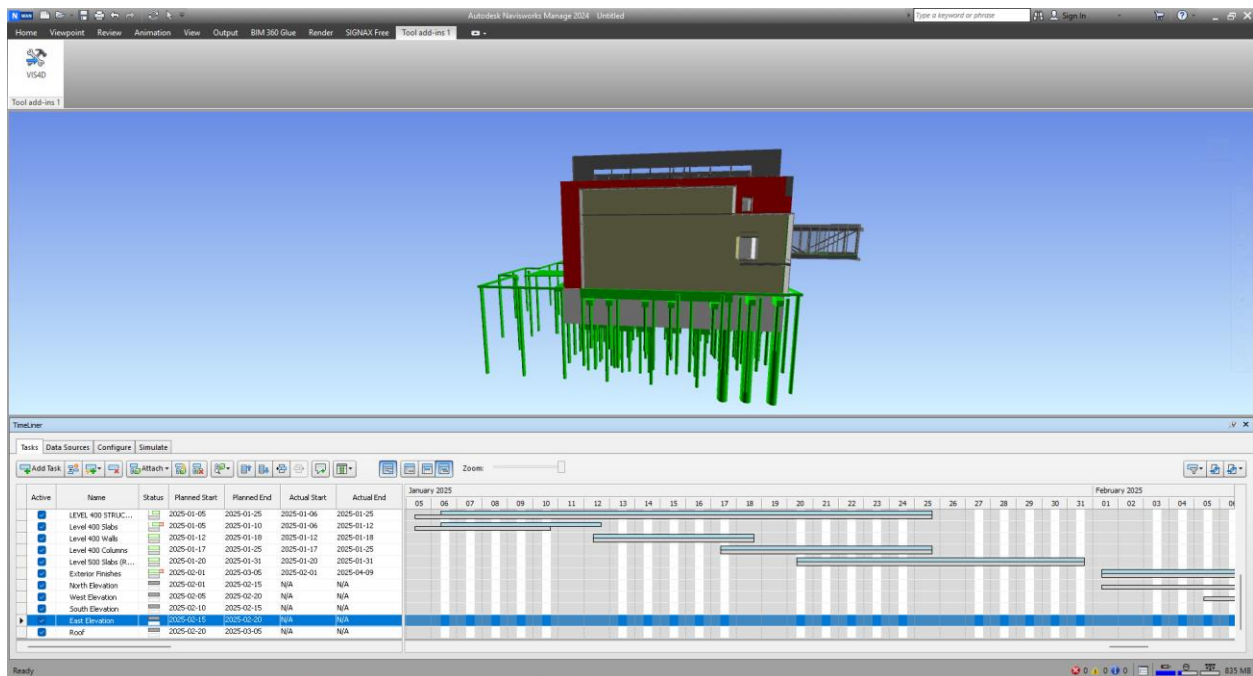


Figure 11. Navisworks schedule after updating east elevation cladding status to on hold using VISA4D

Later in the day, during a routine site walkthrough, the superintendent verifies several active work areas. She confirms that construction work is underway on the north, west, and south elevations. Using the VISA4D mobile app, she updates their statuses via voice: “Set north elevation, west elevation and south elevation to in progress”. The system immediately updates the statuses of all three tasks and changes the colour of the building elements from “Not Started” (grey) to “In Progress” (blue) (Figure 12).

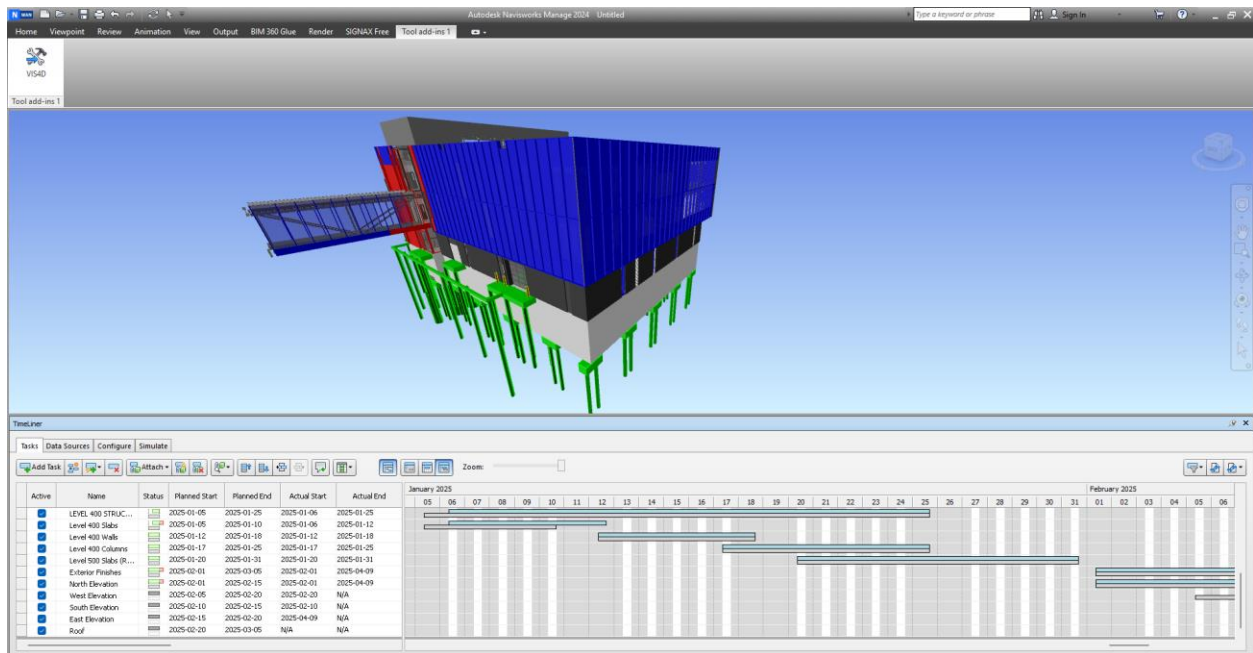


Figure 12. Navisworks schedule after updating the status of west, north and south elevation cladding to in progress using VISA4D

As she approaches the roofing area, she spoke with the crew, who informed her that they would not be able to proceed as planned. In addition to weather issues, the delayed completion of the east elevation prevents their work from starting. The superintendent responds with another voice command: “Update roof installation status to on hold and the roof immediately turns red in the model” (Figure 13).

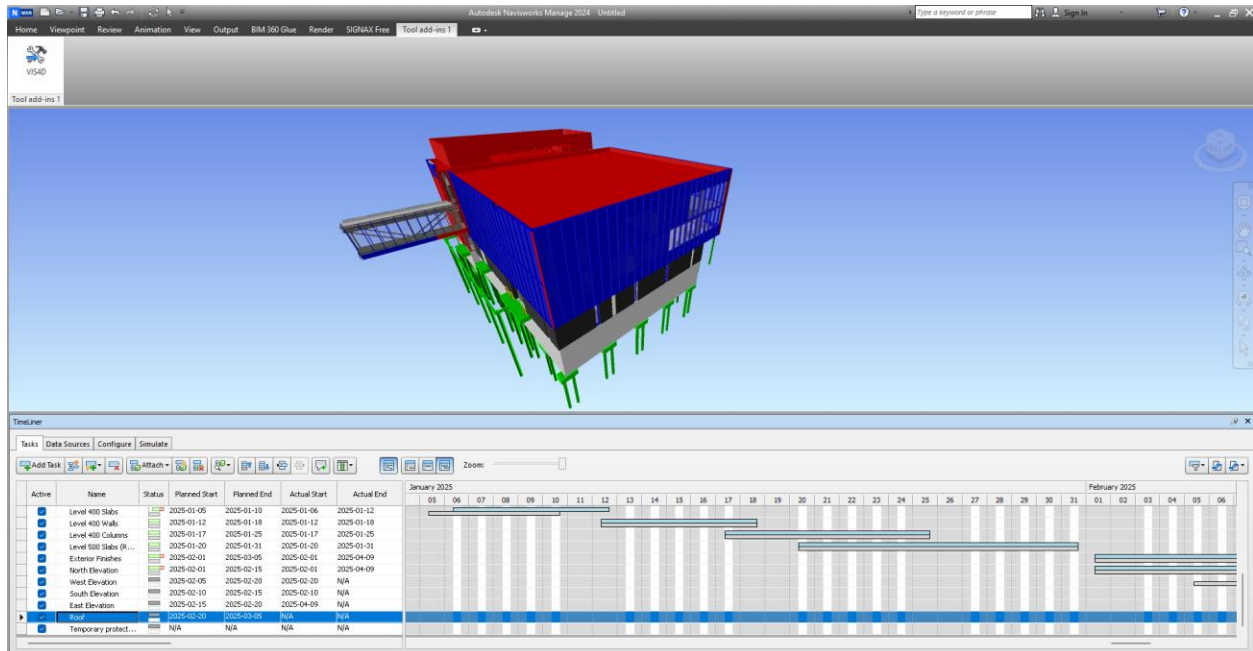


Figure 13. Navisworks schedule after updating the status of roof to on hold using VISA4D

While reviewing scheduled tasks on her device, she notices an outdated task: “Temporary protective canopy installation”. Since this scope was removed after a recent design change, she deletes it using voice input: “Delete temporary protective canopy installation”. The task gets removed from both the schedule and the 4D model in real-time (Figure 14).

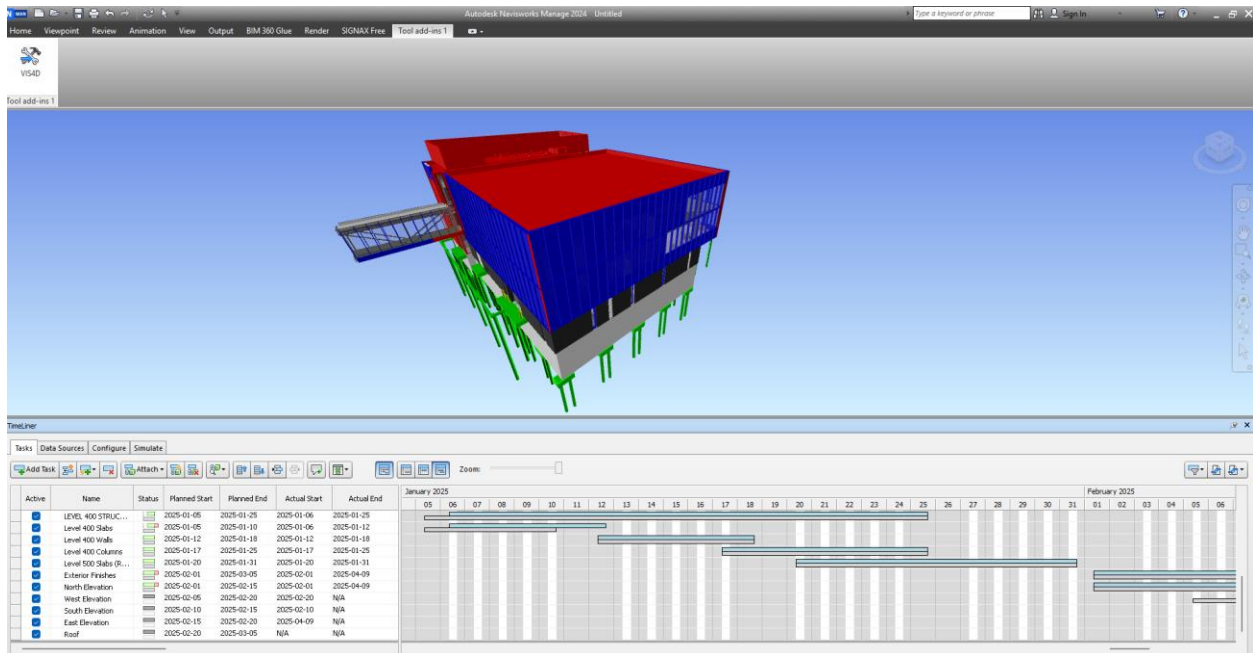


Figure 14. Navisworks schedule after deleting temporary protective canopy installation task using VISA4D

At a project coordination meeting the following day, the Director of Construction Planning reviews the updated model and Gantt chart using VISA4D. Without needing a site visit, he got a clear picture of the project status. In the model (Figure 15), green elements show completed work on all structural components for Levels 100 to 400, including slabs, walls, and columns. Blue indicates in-progress activity on the north, west, and south elevations. Red highlights on hold on the east elevation and roof. Grey shows not started tasks like masonry around the building.

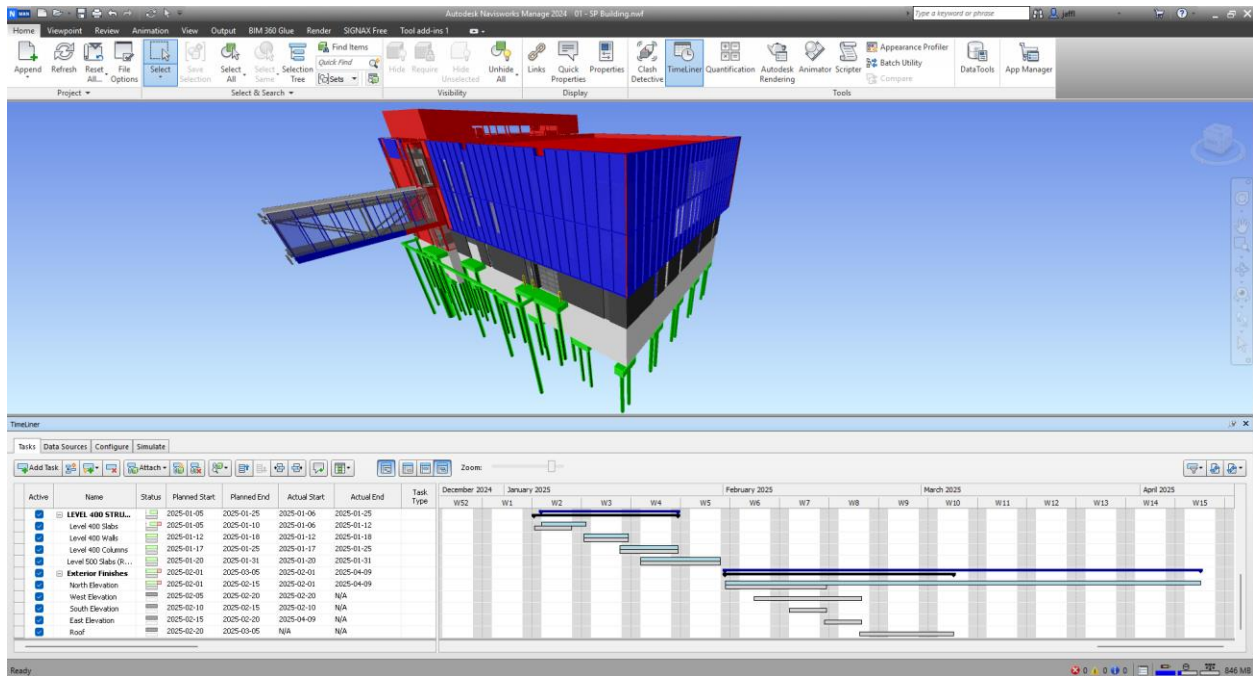


Figure 15. Educational building Navisworks schedule after updating using VISA4D

This immediate visual feedback (Figure 15) allows the director to focus on critical delays and engage with contractors on the next step, minimizing the risk of cost overruns and delays. All schedule data, planned and actual dates, were updated either via text input or voice commands on their mobile devices directly from the site, eliminating paperwork and lag between field activity and office reporting.

Both use cases demonstrate how VISA4D enables more rapid and smarter decision-making by connecting field conditions, model data, and schedule logic, all updated in real-time using voice or text input.

3.4 Validation and Verification of the VISA4D Tool

3.4.1 Validation

As part of the validation process, which in this study refers to user testing aimed at assessing the ease of use and efficiency of the VISA4D tool, the VISA4D tool was tested by 20 undergraduate

and graduate civil engineering students at the University of Manitoba who volunteered to participate in this study. Participant recruitment was conducted in accordance with the approved research ethics protocol, and informed consent was obtained using the consent form provided in [Appendix B](#). Table 2, Figure 16 and Figure 17 show the demographic background of the participants and participants' academic standing and tool experience distribution, respectively.

A real-world office building construction was used as the case study building for the tests, for which a construction schedule was created in Navisworks and linked to its model elements to create the 4D BIM of the project. The user tests were specifically designed to evaluate the ease of use and efficiency of the tool. These included testing multiple activities and primarily focused on adding new tasks, deleting existing tasks, assigning actual dates to tasks, updating or adding task start and finish dates, and modifying the status of tasks. Participants were instructed to engage with at least four random features of the VISA4D tool. During the user testing, all participants tested both voice and text commands.

Table 2. Participant demographics for VISA4D tool validation study

Demographic Category	Number of Participants (N = 20)
Program Level	
Undergraduate Students	6
Graduate Students	14
Year in Program	
3 rd Year	4
4 th Year	2
Graduate (Master's or PhD)	14
Prior Experience with 4D BIM Tools	
Familiar with a 4D BIM tool	13
No prior experience with any 4D BIM tools	7

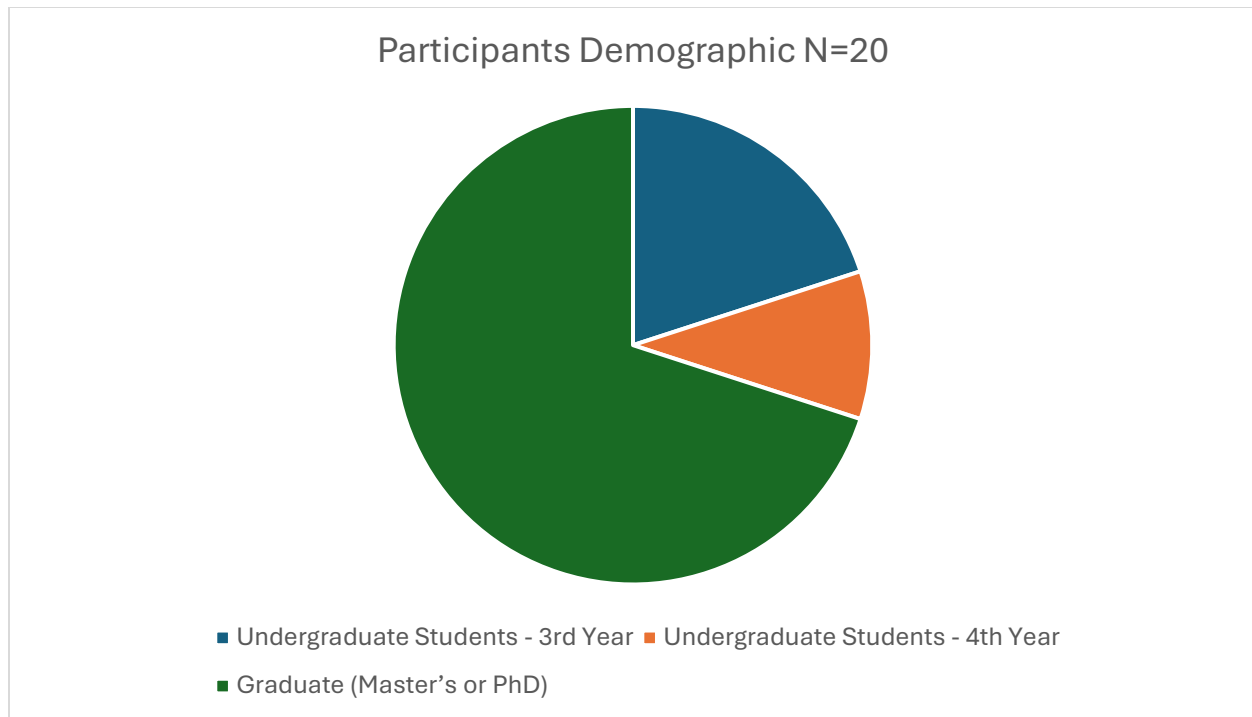


Figure 16. Participants' academic standing distribution

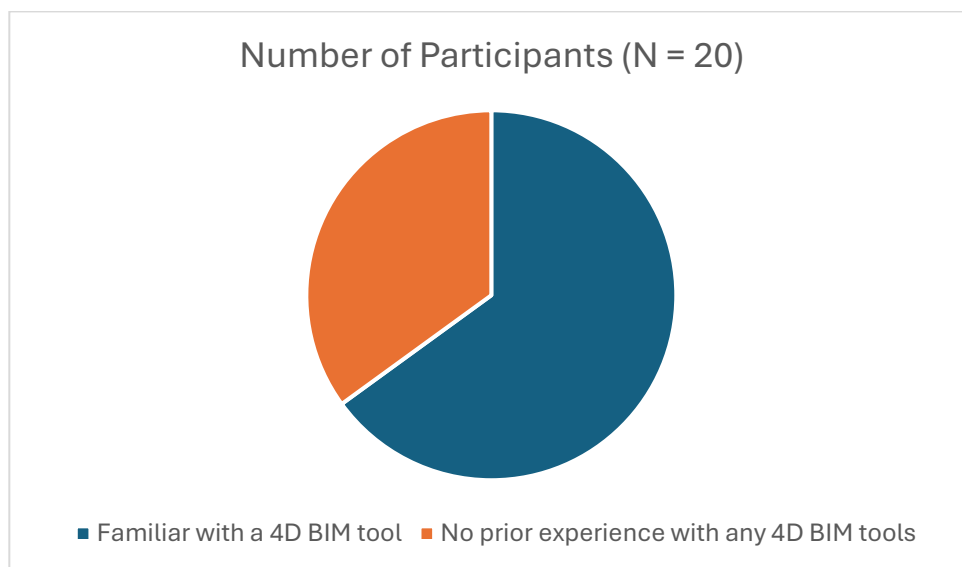


Figure 17. Participants' tool experience distribution

After the hands-on testing, an anonymous survey that included structured multiple-choice questions as well as open-ended questions (presented in Table A1 in [Appendix A](#)) was conducted to gather feedback from the participants, which served as the basis for the validation of the tool. The survey revealed positive outcomes, with 95% of respondents (19 out of 20) reporting that the

tool was very easy or easy to use. Additionally, all participants affirmed that the VISA4D tool consistently or often recognized commands accurately, while 85% reported consistent accuracy. Regarding text handling, all participants found the tool effective, with 85% rating it as very effective.

Voice command handling also yielded favourable results, with 80% of respondents rating this feature as very effective or effective. Furthermore, the tool's response time was considered satisfactory by all participants, with 90% describing it as always satisfactory. The user-friendliness of the interface received high approval, with 85% labeling it as very user-friendly. Learning to operate the tool was deemed very easy by 90% of the testers.

Participants appreciated the simplicity and intuitiveness of the user interface, effective voice control, and visual clarity provided by colour-coding tasks. However, occasional challenges were observed when participants with diverse accents tested the system, as there were difficulties with command recognition. Suggested improvements include enhancing the system's ability to accurately interpret diverse accents and improving overall voice command capabilities. Limitations have been further explained in [Chapter 4](#).

3.4.2 Verification

The reliability and performance of the VISA4D tool were investigated and verified through a systematic testing approach focusing on intent classification capabilities. In this context, verification refers to testing how accurately the system interprets and processes user commands, ensuring that predicted intents and extracted task names match the expected outcomes. A custom testing framework was developed in Python to assess the precision of the NLP processor class in interpreting and responding to a range of construction-specific commands.

To achieve broad coverage and assess the reliability of the test framework, each of the five intent categories (i.e., create task, delete task, update date, update status, and unknown) was evaluated with a set of 20 unique commands, resulting in a total of 80 commands tested. Table 3 presents the sample queries categorized by their functional perspectives. This collection was designed to encompass a wide array of the prototype’s potential use cases, facilitating a thorough evaluation of its functional accuracy.

Table 3. Sample questions categorized by their perspective commands

Commands	Sample Queries	
Create task	Add Foundation Piling task	Can you create Electrical Wiring
	Create a new task for Grade Beams Pour #3	Please add Plumbing Installation
	Set up Elevator Raft	Add task Exterior Siding
	I need a new task for Feature Wall Pours	Create a new task for Door 2000 Installation
	Can you create Main Floor Glazing Installation	Set up Stair 1000
	Please add Second Floor Drywall Installation	I need a new task for Slab A
	Add task Roof Door Installation	Can you create Structural Framing
	Create a new task for Concrete Pouring	Please add Final Inspection
	Set up HVAC Installation	Add task Grade Beams Pour #5
Update Status	I need a new task for Window Installation	Create a new task for Main Floor Door Installation
	Update the status of Foundation Piling to complete	Update the status of Electrical Wiring to in progress
	Change Grade Beams Pour #3 status to in progress	Change Plumbing Installation status to complete
	Set Elevator Raft as on hold	Set Exterior Siding as in progress
	Mark Feature Wall Pours complete	Mark Door 2000 Installation complete
	Update Main Floor Glazing Installation to in progress	Update Stair 1000 to on hold
	Can you update Second Floor Drywall Installation to on hold	Can you update Slab A to in progress
	Please change Roof Door Installation status to complete	Please change Structural Framing status to complete
	Update Concrete Pouring to in progress	Update Final Inspection to on hold

	Set HVAC Installation as complete	Set Grade Beams Pour #5 as in progress
	Mark Window Installation on hold	Mark Main Floor Door Installation complete
Delete Task	Delete Foundation Piling task	Delete the Electrical Wiring task
	Remove Grade Beams Pour #3 task	Could you please remove Plumbing Installation task
	Delete task Elevator Raft from the TimeLiner	I'd like to remove Exterior Siding
	Remove task painting	Please remove Door 2000 Installation
	Delete Main Floor Glazing Installation	Would you delete Stair 1000 task for me
	Can you remove Second Floor Drywall Installation	Can you remove task Slab A
	Delete task Roof Door Installation	I need to terminate Structural Framing
	Remove the Concrete Pouring task	Please remove the Final Inspection task
	I would like you to remove HVAC Installation	Delete the task Grade Beams Pour #5
	Remove Window Installation	I want you to cancel Main Floor Door Installation
Update Date	Update start date for Foundation Piling to April 5	Update start date for Electrical Wiring to March 28
	Update finish date for Grade Beams Pour #3 to May 12	Update finish date for Plumbing Installation to July 18
	Update start date for Elevator Raft to June 20	Update start date for Exterior Siding to May 20
	Update end date for Feature Wall Pours to July 10	Update end date for Door 2000 Installation to June 28
	Update start date for Main Floor Glazing Installation to March 15	Update start date for Stair 1000 to April 10
	Update end date for Second Floor Drywall Installation to August 22	Update finish date for Slab A to August 5
	Update start date for Roof Door Installation to April 30	Update start date for Structural Framing to March 25
	Update finish date for Concrete Pouring to May 8	Update end date for Final Inspection to September 30
	Update start date for HVAC Installation to June 15	Update finish date for Grade Beams Pour #5 to April 22
	Update end date for Window Installation to September 3	Update start date for Main Floor Door Installation to May 15

Each intent category was tested with tailored command types:

- Task creation: to test the ability of the system to recognize requests to add new construction tasks.
- Status update: to verify accuracy in recognizing status change requests.
- Date update: to test the system’s ability to recognize different date patterns and distinguish between start and end date modifications
- Task deletion: to test the system’s ability to recognize task removal requests.

These commands were stored in a structured CSV file, with command sets designed using varied phrasings, orders, and construction-specific terminology to reflect real-world language use and test the robustness of the NLP processor class. To handle such variations, VISA4D employs several techniques: the NLP processor uses semantic similarity evaluation to interpret differently phrased commands with similar meanings, as shown in [Chapter 3, section 3.2.2](#), where phrases like “start concrete pouring” and “begin concrete work” are understood as equivalent. For ambiguous expressions, the system applies a BERT-based model to map text into a vector space for comparison and, when confidence scores fall below set thresholds, it prompts the user for clarification, ensuring accurate interpretation even when construction professionals use unclear terms.

The framework analyzed each command by comparing the predicted task name and intent with the expected intent and expected task name. Then, each command was fed into the NLP processor class, and the responses were captured and analyzed, extracting predicted intents and task names.

Finally, the accuracy metrics were computed by comparing predicted values against expected values. Python libraries, including *pandas* for data manipulation, *sklearn.metrics* for

classification metrics, and *matplotlib* with *seaborn* for visualization of results, were used. Various performance metrics were computed and visualized:

- Intent accuracy: percentage of commands where the predicted intent matched the expected intent.
- Task name extraction accuracy: percentage of commands where the extracted task name correctly matched the expected task name.

Figure 18 shows the accuracy results across command types. While most intent categories demonstrated high performance, task creation and date updates showed slightly lower task accuracy, pointing to some confusion in those areas.

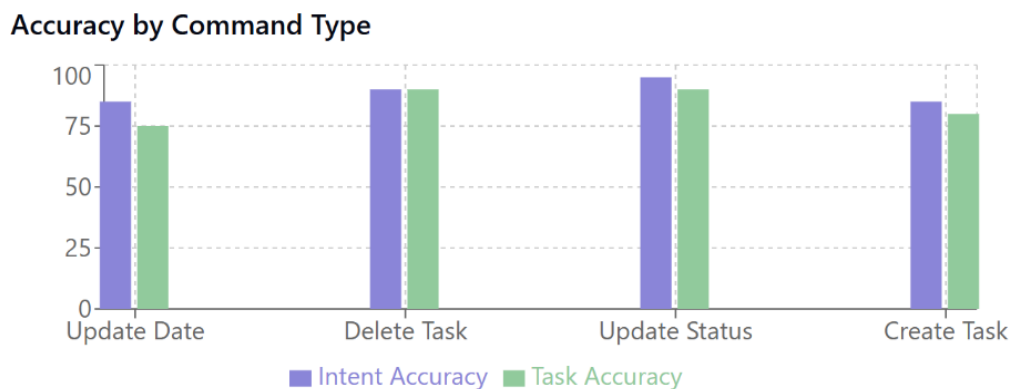


Figure 18. VISA4D accuracy analysis

A total of nine errors were identified, as detailed in Figure 19. The most common misclassifications included:

- “Update date” commands misclassified as “update status” (three cases);
- “Unknown” commands interpreted as “create task” (three cases);
- “Delete task” commands misclassified as “create task” (two cases);
- One “update status” command classified as unknown.

Error Distribution (9 Total Errors)

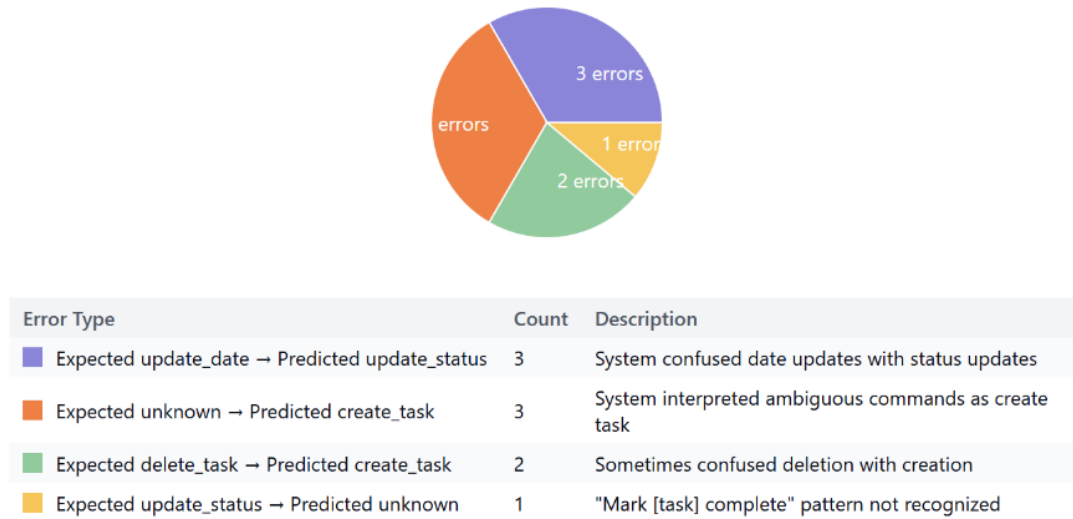


Figure 19. Error distribution

To facilitate comprehensive analysis, a confusion matrix was employed, as shown in Figure 20, providing a clear representation of both correctly and incorrectly classified instances.

Expected Intent	Predicted Intent					Total
	Create Task	Delete Task	Unknown	Update Date	Update Status	
Create Task	17	0	0	0	0	17
Delete Task	2	18	0	0	0	20
Unknown	3	0	0	0	0	3
Update Date	0	0	0	17	3	20
Update Status	0	0	1	0	19	20
Total	22	18	1	17	22	80
	Correct Classifications					
	Incorrect Classifications					

Figure 20. Confusion matrix

The confusion matrix visualization clearly differentiates correct classifications, represented by green diagonal cells, from incorrect classifications, highlighted in red. The intensity of the colours reflects the frequency of occurrences to provide a visual insight into areas of strength and weakness of the tool. The confusion matrix revealed specific error patterns, such as three instances (15%) of “update_date” command incorrectly classified as “update_status”, three ambiguous or “unknown”

commands misclassified as “create_task”, and one instance (5%) where the “update_status” command was incorrectly recognized as “unknown”.

The system achieved an overall accuracy of 89%, successfully classifying 71 out of the 80 tested commands. Performance varied across intent types, highlighting specific strengths and challenges. The detailed evaluation given in Table 4 shows the precision, recall, F1-score (the F1 score represents the mean between precision and recall), and support metrics, which further clarified the system’s performance across individual intent types.

Table 4. Detailed evaluation of intent

Intent	Precision (%)	Recall (%)	F1-Score	Support
Create Task	77	100	87	17
Delete Task	100	90	95	20
Unknown	0	0	0	3
Update Date	100	85	92	20
Update Status	86	95	91	20

The “create_task” intent demonstrated perfect recall (100%) but lower precision (77%), primarily due to the confusion and misclassification of ambiguous commands, resulting in an F1-score of 87%. Conversely, the “delete_task” intent exhibited flawless performance with 100% precision, recall, and an F1-score of 100%, reflecting its high reliability. The “unknown” intent faced significant challenges, yielding zero precision and recall, indicating the system’s inability to correctly classify ambiguous commands, resulting in an F1-score of 0%.

For the “update_date” intent, the system achieved perfect precision (100%) but slightly reduced recall (85%), leading to an F1-score of 92%, influenced by occasional misclassification as “update_status”. Lastly, the “update_status” intent showed robust performance, with high recall (95%) and considerable precision (86%), resulting in an F1-score of 91%.

In conclusion, while the system demonstrates strong overall capability in intent recognition, the detailed analysis highlighted key areas for further improvement. Specifically,

future developments should focus on enhancing the system’s ability to manage ambiguous inputs and more accurately distinguish closely related intents such as “update_date” and “update_status”. These refinements will further increase accuracy, reliability, and operational effectiveness in practical applications.

Chapter 4: Discussion

4.1 Discussion

The VISA4D framework presents a transformative step in the integration of conversational AI with 4D scheduling to support construction schedule updates and maintenance. The developed tool leverages NLP techniques and enables users to interact with the 4D BIM environment through voice or text commands, reducing the need for specialized technical expertise. This allows on-site personnel to update task details, such as dates, statuses, and interdependencies, directly from mobile devices on daily basis. The interface is designed to be intuitive and user-friendly, particularly for users with limited or no experience using tools like Navisworks. When Navisworks is reopened, updates are automatically synchronized, ensuring that the project files remain up-to-date. This reduces administrative effort, supports real-time decision making, and improves communication among project team members. Moreover, the tool’s accessibility helps broaden participation in schedule management and reduces the learning curve, typically associated with BIM tools. In addition, its modular and scalable structure supports VISA4D’s application across a range of project sizes and complexities, making it a practical contribution to digital construction workflows.

The validation and verification results further support these benefits. User tests demonstrated high usability ratings, with 95% finding the tool easy to use and all participants

confirming accurate command recognition. Additionally, the systematic verification testing revealed an overall system accuracy of 89% across different command types, with particularly strong performance in task deletion (100% accuracy) and task status updates (91% F1-score). These findings validate the practical effectiveness of the VISA4D framework for construction schedule updating and maintenance processes.

While the current implementation primarily interfaces with Navisworks' TimeLiner functionality, VISA4D's modular architecture is designed with extensibility in mind. This approach would allow for adaptation to support different scheduling methodologies, including pull scheduling and other lean construction techniques. The system's remote access capabilities and ability to facilitate operational adjustments to project schedules make it particularly well suited for integration with lean construction tools.

4.2 Limitations

While the validation and verification results presented in [Chapters 3, Section 3.4](#) demonstrate promising potential for the VISA4D tool, several important limitations and challenges must be acknowledged for a comprehensive evaluation of the system's capabilities and constraints. The future work that emerges from these identified limitations is detailed in [Chapter 5](#).

4.2.1 Sample Size and Participant Demographics

The user testing involved 20 undergraduate and graduate civil engineering students from the University of Manitoba, which presents certain limitations. First, the relatively small sample size might not adequately reflect the wider community of potential users. Second, all participants come from an academic background with similar educational training, which may not accurately reflect how construction professionals with varied expertise and technical backgrounds would interact

with the tool. Construction professionals, especially those with extensive field experience but limited exposure to advanced digital tools, might face different usability challenges than those encountered by engineering students who are typically more familiar with digital interfaces and BIM software.

Additionally, the testing was conducted in a controlled academic setting, which differs substantially from actual construction project environments. Real-world deployment would introduce additional variables that were not captured in the validation study.

4.2.2 Speech Recognition Challenges

As evidenced in both the user testing feedback and technical evaluation, speech recognition presented notable challenges. Participants with diverse accents reported occasional difficulties with command recognition, which aligns with the limitations identified in our technical assessment. The current speech recognition system exhibited significant recognition gaps for construction-specific terminology such as “grade beams”, “HVAC”, “MEP systems”, and specialized identifiers like “concrete pour #3”, which fall outside everyday language datasets on which most speech recognition models are trained.

Although our implementation of a construction-specific vocabulary enhancement within the NLP processor component improved recognition accuracy, this approach has inherent limitations, as it requires manual definition of construction terminology and does not automatically adapt to project-specific terminology. Additionally, construction terminology varies significantly across regions and projects, creating recognition challenges that would require continual system refinement.

The environmental conditions of typical construction sites also present practical challenges for speech recognition. Construction environments are often extremely loud, which would

significantly impact speech recognition accuracy and may require users to repeat commands or prefer text input in noisy conditions over voice interaction.

4.2.3 Technical Infrastructure Constraints and Integration Challenges

The current VISA4D architecture was tested primarily in individual user sessions but may face scalability limitations when deployed on larger construction projects. The local HTTP server implementation, while effective for individual Navisworks sessions, could encounter performance bottlenecks when handling multiple simultaneous users. Construction projects typically require numerous team members to concurrently access and update schedule information, necessitating additional concurrency controls to manage conflicting modifications.

Moreover, the system's reliance on *OAuth* for Autodesk Platform Services introduces dependency on internet connectivity for authentication and token renewal. While the Task Manager component implements local state caching to provide some offline functionality, extended periods without connectivity would render the system unresponsive. This limitation could be particularly problematic in remote construction sites with limited or intermittent internet access.

Finally, long-term viability requires continuous maintenance to preserve compatibility with evolving Navisworks platforms and API changes, representing an ongoing development challenge for the system.

4.2.4 Intent Recognition Accuracy Implications

The technical evaluation revealed specific weaknesses in intent recognition that warrant consideration for practical deployment. The “create_task” intent demonstrated perfect recall (100%) but lower precision (77%), indicating that the system occasionally misclassified other commands as task creation requests. In a real-world construction management context, such

misclassifications could lead to unintended task additions, potentially introducing errors into project schedules that might go unnoticed until they impact workflow or resource allocation.

The system also performed poorly with “unknown” intents, yielding zero precision and recall, which reflects an inability to properly flag ambiguous commands. This limitation could lead to confusion in practical settings, where users might believe a command was successfully executed when, in fact, it was misinterpreted by the system. The confusion between closely related intents such as “update_date” and “update_status” (where three “update_date” commands were misclassified as “update_status”) could result in incorrect schedule information being recorded.

Chapter 5: Conclusions and Recommendations

5.1 Summary and Conclusions

This study presents VISA4D, a novel framework that uses voice and text commands and AI algorithms to streamline real-time construction schedule updating and maintenance within 4D BIM environments. At its core, VISA4D integrates an AI-driven chatbot with Autodesk Navisworks, thereby enabling users to efficiently and practically generate, update, and maintain construction schedules. It addresses the limitations of traditional schedule updating processes in construction projects by integrating conversational AI with a 4D BIM tool, and bridges the gap between field operations and digital systems in the context of construction schedule updating and maintenance.

The VISA4D framework, as demonstrated through user testing, has the potential to simplify the complexities associated with 4D BIM operations. By making construction schedule creation and updating more accessible and intuitive, the system can empower a wider range of construction professionals to engage with 4D BIM tools. Its user-friendly approach promotes a

more interactive and flexible method for managing construction schedules, ultimately supporting better collaboration and communication within construction project teams.

With the ongoing digital transformation of the industry, employing such advanced technologies is critical for improving the efficiency of construction projects while fostering collaboration and enhancing the overall quality of the project outcomes. In that sense, VISA4D represents a substantial leap forward in the realm of digital construction technologies and illustrates the potential for AI integration in construction. By facilitating direct, real-time engagement with 4D BIM models, it establishes a new benchmark for construction scheduling and project management.

5.2 Recommendations for Future Work

As outlined in [Chapter 4](#), the current version of the VISA4D framework has several limitations that necessitate further investigation and refinement. Future work should focus on addressing these limitations through several key initiatives. First, conducting expanded validation with industry professionals would provide valuable insights into performance and usability in real-world construction settings. Similarly, extending capability beyond Autodesk Navisworks to other 4D BIM platforms would enhance the tool's applicability across different project environments. This could be achieved by introducing a platform-agnostic integration layer, leveraging open standards such as Industry Foundation Classes 4D (IFC 4D) [55], which is an extension of the Industry Foundation Classes format that combines 3D geometry with scheduling data, and developing dedicated connectors to translate interpreted commands into the APIs of tools like Synchro or Bentley. By decoupling the NLP processor from the Navisworks, VISA4D could maintain consistent functionality across diverse platforms, thereby increasing its value in varied project contexts.

Developing adaptive learning techniques for the NLP component could improve recognition of construction-specific terminology without the need for manual configuration. This includes expanding the command vocabulary, improving context-awareness, and enabling automatic incorporation of new terms.

Future development should also address the scalability challenges by implementing robust concurrency controls and optimizing the system architecture for multi-user scenarios. Improvements to error handling and recovery mechanisms would also increase system resilience. Additionally, incorporating multilingual capabilities in future versions would broaden accessibility, allowing users to interact with the system in their preferred language. This feature would be especially valuable in global projects, where stakeholders, contractors, and team members often speak different languages, ensuring smoother collaboration and reducing the risk of miscommunication. Another enhancement could include a track-changes feature, enabling users to log and review past modifications for auditability and better project documentation. Finally, implementing privacy-preserving techniques, along with noise-resistant speech recognition algorithms, would mitigate security risks and improve performance in the noisy environments typical of construction sites.

Contribution to Literature

This master's thesis represents the research conducted between 2023 and 2025 within the Construction Engineering and Management Research Group, Department of Civil Engineering, University of Manitoba, Canada. The following publications are the contributions to knowledge:

1. Jaff, L.; Guven, G. Implementing AI to Improve the Efficiency of BIM-based Construction schedules. Accepted and presented at the Canadian Society for Civil Engineering (CSCE), Niagara Falls, Ontario, Canada, 5-7 June 2024.
2. Jaff, L.; Garg, S.; Guven, G. A Novel Framework for Natural Language Interaction with 4D BIM. Buildings 2025, 15, 1840. <https://doi.org/10.3390/buildings15111840>
3. Jaff, L.; Garg, S.; Guven, G. AI-supported real-time schedule updating and maintenance. Accepted and presented at the Joint Canadian Society for Civil Engineering Construction Specialty and Construction Research Congress (CSCE-CRC), Montreal, Quebec, Canada, 28-31 July 2025.

References

1. Sacks, R.; Eastman, C.; Lee, G.; Teicholz, P. *BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors*, 3rd ed.; Wiley: Hoboken, NJ, USA, 2018.
2. Staub-French, S.; Khanzode, A. 3D and 4D modeling for design and construction coordination: issues and lessons learned, *J. IT. Constr.* **2007**, *Vol 12*, pp. 381-407. [[CrossRef](#)]
3. Mazars, T.; Adel, F. Chronographical spatiotemporal dynamic 4D planning. *Autom. Constr.* **2020**, *Vol 112*, 103076. [[CrossRef](#)]
4. Doukari, O.; Seck, B.; Greenwood, D. The efficient generation of 4D BIM construction schedules: a case study of the Nanterre 2 CESI project in France. *Frontiers in Built Environment* **2022**, *Vol 8*. [[CrossRef](#)]
5. Doukari, O.; Seck, B.; Greenwood, D. The creation of construction schedules in 4D BIM: A comparison of conventional and automated approaches. *Buildings* **2022**, *Vol 12*, 1145. [[CrossRef](#)]
6. Nguyen, T. BIM in construction: benefits, challenges, and development trends. *International Journal of Scientific Research in Science, Engineering and Technology*. **2025**, *Vol 12*, pp. 160-165. [[CrossRef](#)]
7. Ahmadi, P.; Arashpour, M. An analysis of 4D-BIM construction planning: Advantages, risks and challenges. In Proceedings of 37th International Symposium on Automation and Robotics in Construction (ISARC), Kitakyushu, Japan, 27-28 October 2020. [[CrossRef](#)]
8. Moon, H.; Kim, H.; Kim, C.; Kang, L. Development of a schedule-workspace interference management system simultaneously considering the overlap level of parallel schedules and workspaces. *Autom. Constr.* **2014**, *Vol 39*, pp. 93-105. [[CrossRef](#)]

9. Mayouf, M.; Jones, J.; Elghaish, F.; Emam, H.; Ekanayake, E.M.A.C.; Ashayeri, I. Revolutionising the 4D BIM process to support scheduling requirements in modular construction. *Sustainability*. **2024**, *Vol* 16, pp. 476. [[CrossRef](#)]
10. Milat, M.; Knezić, S.; Sedlar, J. Resilient scheduling as a response to uncertainty in construction projects. *Applied Sciences* **2021**, *vol* 11. 6493. [[CrossRef](#)]
11. Kiziltas, S.; Akinci, B. The need for prompt schedule update by utilizing reality capture technologies: A case study. In Proceedings of the Construction Research Congress: Broadening Perspectives, San Diego, California, USA, 5-7 April 2005. [[CrossRef](#)]
12. Das, S.; Das, D. Natural Language Processing (NLP) techniques: Usability in human-computer interactions. In Proceedings of the 2024 6th International Conference on Natural Language Processing (ICNLP), Xi'an, China, 22-24 March 2024. [[CrossRef](#)]
13. Botton, C.; Kubicki, S.; Halin, G. 4D/BIM Simulation for pre-construction and construction scheduling. multiple levels of development within a single case study. In Proceedings of Creative Construction Conference, Kraków, Poland, 21-24 June 2015. [[CrossRef](#)]
14. Zhou, W.; Whyte, J.; Sacks, R. Construction safety and digital design: a review. *Autom. Constr.* **2012**, *vol* 22, pp. 102–11. [[CrossRef](#)]
15. Sloot, R. N. F.; Heutink, A.; Voordijk, J. T. Assessing usefulness of 4D BIM tools in risk mitigation strategies. *Autom. Constr.* **2019**, *Vol* 106, 102881. [[CrossRef](#)]
16. Sciuto, A.; Saini, A. Forlizzi, J.; Hong, J. Hey Alexa, what's up?: A mixed-methods studies of in-home conversational agent usage. In Proceedings of the 2018 Designing Interactive Systems Conference (DIS '18), Hong Kong, China, 9-13 June 2018. [[CrossRef](#)]
17. Hoy, M. B. Alexa, Siri, Cortana, and more: An introduction to voice assistants. *Medical Reference Services Quarterly* **2018**. *Vol.* 37, pp. 81–88. [[CrossRef](#)]

18. Bello, S.; Oyedele, L.; Akanbi, L.; Bello, A.-L. Cloud computing for chatbot in the construction industry: An implementation framework for conversational-BIM voice assistant. *Digital Engineering* **2025**, Vol 5, pp. 100031. [[CrossRef](#)]
19. Son, H.; Kim, C.; Kwon Cho, Y. Automated schedule updates using as-built data and a 4D building information model. *Journal of Management in Engineering* **2017**, Vol. 33. [[CrossRef](#)]
20. Wang, X.; Truijens, M.; Hou, L.; Wang, Y.; Zhou, Y. Integrating augmented reality with building information modeling: onsite construction process controlling for liquefied natural gas industry. *Autom. Constr.* **2014**, Vol 40, pp. 96-105. [[CrossRef](#)]
21. Trebbe, M.; Hartmann, T.; Dorée, A. 4D CAD models to support the coordination of construction activities between contractors.” *Autom. Constr.* **2015**, Vol 39, pp. 83-91. [[CrossRef](#)]
22. Jaff, L.; Garg, S.; Guven, G. AI-supported real-time schedule updating and maintenance. In the Proceedings of Joint Canadian Society for Civil Engineering Construction Specialty and Construction Research Congress (CSCE-CRC), Montreal, Quebec, Canada, 28-31 July 2025.
23. Asadzadeh, A.; Arashpour, M.; Li, H.; Ngo, T.; Bab-Hadiashar, A.; Rashidi, A. Sensor-based safety management. *Autom. Constr.* **2020**, Vol 113, 103128. [[CrossRef](#)]
24. Amer, F.; Koh, H. Y.; Golparvar-Fard, M. Automated methods and systems for construction planning and scheduling: critical review of three decades of research. *Journal of Construction Engineering and Management* **2021**, Vol. 147. [[CrossRef](#)]
25. Prieto, S. A.; Mengiste, E. T.; García de Soto, B. Investigating the use of ChatGPT for the scheduling of construction projects. *Buildings* **2023**, Vol 13, 857. [[CrossRef](#)]

26. Nabavi, A.; Ramaji, I.; Sadeghi, N.; Anderson, A. Leveraging natural language processing for automated information inquiry from building information models. *Journal of Information Technology in Construction* **2023**, Vol 28, pp. 266-285. [[CrossRef](#)]
27. Singh, K.; Pal, A.; Kumar, P.; Lin, J.; Hsieh S.-H. Prospects of integrating BIM and NLP for automatic construction schedule management. In the proceedings of the 40th International Symposium on Automation and Robotics in Construction, IAARC, Chennai, India, 5-7 July 2023. [[CrossRef](#)]
28. Aljebory, K. M.; QaisIssam, M. Developing AI based scheme for project planning by expert merging Revit and Primavera software. In Proceedings of the 16th International Multi-Conference on Systems, Signals & Devices (SSD), Istanbul, Turkey, 21-24 March 2019. [[CrossRef](#)]
29. Gado, N. G. AI revolutionizes construction management “Building smarter, safer, and efficiently addressing industry challenges”. *Engineering Research Journal* **2024**, Vol 183, pp. 330-344. [[CrossRef](#)]
30. Khobragade, A.N.; Maheswari, N.; Sivagami, M. Analyzing the housing rate in a real estate informative system: a prediction analysis. *International Journal of Civil Engineering and Technology* **2018**, Vol 9, pp. 1156-1164. [[CrossRef](#)]
31. Poh, C. Q. X.; Ubeynarayana, C. U.; & Goh, Y. M. Safety leading indicators for construction sites: A machine learning approach. *Autom. Constr.* **2018**, Vol 93, pp. 375-386. [[CrossRef](#)]
32. Ajayi, A.; Oyedele, L.; Owolabi, H.; Akinade, O.; Bilal, M.; Davila Delgado, J. M.; Akanbi, L. Deep learning models for health and safety risk prediction in power infrastructure projects. *Risk Analysis* **2020**, Vol 40, pp. 2019–2039. [[CrossRef](#)]

33. Barreiro, J.; Boyce, M.; Do, M.; Frank, J.; Iatauro, M.; Kichkaylo, T.; Morris, P.; Ong, J.; Remolina, E.; Smith, T. EUROPA: A platform for AI planning, scheduling, constraint programming, and optimization. In Proceedings of the International Conference on Automated Planning and Scheduling, Atibaia, São Paulo, Brazil, 25-29 June 2012. [[CrossRef](#)]
34. Lin, W. Y. Prototyping a chatbot for site managers using building information modeling (BIM) and natural language understanding (NLU) techniques. *Sensors* **2023**, *Vol 23*, 2942. [[CrossRef](#)]
35. Singh, J.; Anumba, C. J. Real-time pipe system installation schedule generation and optimization using artificial intelligence and heuristic techniques. *Journal of Information Technology in Construction* **2022**, *Vol 27*, pp. 173–190. [[CrossRef](#)]
36. Ma, Y.; Qiao, E. Visualized management system of prefabricated construction and installation engineering based on BIM and machine learning. In the Proceedings of IEEE International Conference on Industrial Application of Artificial Intelligence (IAAI), Harbin, China, 24-26 December 2021. [[CrossRef](#)]
37. Zheng, J.; Fischer, M. Dynamic prompt-based virtual assistant framework for BIM information search. *Autom. Constr.* **2023**, *Vol 155*, 105067. [[CrossRef](#)]
38. You, H.; Ye, Y.; Zhou, T.; Zhu, Q.; Du, J. Robot-enabled construction assembly with automated sequence planning based on ChatGPT: RoboGPT. *Buildings* **2023**, *Vol 13*, 1772. [[CrossRef](#)]
39. Fernandes, D.; Garg, S.; Nikkel, M.; Guven, G. A GPT-powered assistant for real-time interaction with Building Information Models, *Buildings* **2024**, *Vol 14*, 2499. [[CrossRef](#)]

40. Golparvar-Fard, M.; Peña-Mora, F.; Savarese, S. Automated progress monitoring using unordered daily construction photographs and IFC-based Building Information Models. *Journal of Computing in Civil Engineering* **2015**, Vol 29. [[CrossRef](#)]
41. Liu, S.-S.; Shih, K.-C. Construction rescheduling based on a manufacturing rescheduling framework. *Autom. Constr.* **2009**, Vol 18, pp. 715-723 [[CrossRef](#)]
42. Rebolj, D.; Babič, N. Č.; Magdič, A.; Podbreznik, P.; Pšunder, M. Automated construction activity monitoring system. *Autom. Constr.* **2008**, Vol 22, pp. 493-503. [[CrossRef](#)]
43. Kim, C.; Son, H.; Kim, C. Automated construction progress measurement using a 4D Building Information Model and 3D data. *Autom. Constr.* **2013**, Vol 31, pp. 75-82 [[CrossRef](#)]
44. Peffers, K.; Tuunanen, T.; Rothenberger, M.; Chatterjee, S. A design science research methodology for information systems research. *Journal of Management Information Systems* **2007**, Vol 24, pp. 45-77. [[CrossRef](#)]
45. Venable, John & Baskerville, R. Eating our own cooking: Toward a more rigorous design science of research methods. *Electronic Journal of Business Research Methods* **2012**, Vol 10, pp. 141-153. [[CrossRef](#)]
46. Venable, J.; Pries-Heje, J.; Baskerville, R. FEDS: A framework for evaluation in design science research. *European Journal of Information Systems* **2016**, Vol 25, pp. 77-89. [[CrossRef](#)]
47. Hevner, A. a three cycle view of design science research. *Scandinavian Journal of Information Systems* **2007**, Vol 19. [[CrossRef](#)]
48. Brocke, J.; Winter, R.; Hevner, A.; Maedche, A. Accumulation and evolution of design knowledge in design science research – A journey through time and space. *Journals of the Association for Information Systems* **2020**, Vol 21, pp. 520-544. [[CrossRef](#)]

49. Gregor, S.; Hevner, A. Positioning and presenting design science research for maximum impact. *MIS Quarterly* **2013**, *Vol 37*, pp. 337-356. [[CrossRef](#)]
50. Baskerville, R.; Baiyere, A.; Gregor, S.; Hevner, A.; Rossi, M. Design science research contributions: finding a balance between artifact and theory. *Journal of the Association for Information Systems* **2018**, *Vol 19*, pp. [[CrossRef](#)]
51. Sein, M.; Henfridsson, O.; Purao, S.; Rossi, M.; Lindgren, R. Action design research. *MIS Quarterly* **2011**, *Vol 35*, pp. 37-56. [[CrossRef](#)]
52. Sonnenberg, C.; Brocke, J. Evaluations in the Science of the Artificial – Reconsidering the Build-Evaluate Pattern in Design Science Research. *Lecture Notes in Computer Science LNCS* **2012**, pp. 381-397. [[CrossRef](#)]
53. Prat, N.; Wattiau, I.; Akoka, J. Artifact evaluation in information systems design science research? A holistic view. In the proceedings of Pacific Asia Conference on Information Systems, PACIS, Chengdu, China, 24-28 June 2014. [[CrossRef](#)]
54. Jaff, L.; Garg, S.; Guven, G. Voice-Integrated-Schdueling-Assistant-for-4D-BIM-VISA4D, *GitHub*, <https://github.com/SGradient/Voice-Integrated-Schdueling-Assistant-for-4D-BIM-VISA4D>- 2025. [[CrossRef](#)]
55. BuildingSMART International. (2023). Industry Foundation Classes (IFC). Retrieved from <https://technical.buildingsmart.org/standards/ifc>

APPENDIX

APPENDIX A

Table A1. Validation questions for participants

Questions	Options/Response
Q1: Ease of use of VISA4D Tool	The tool was very easy to use The tool was easy to use The tool was neither easy nor difficult to use The tool was difficult to use The tool was very difficult to use
Q2: Command recognition of the VISA4D Tool	The tool consistently recognized my commands accurately The tool often recognized my commands accurately The tool sometimes recognized my commands accurately The tool rarely recognized my commands accurately The tool never recognized my commands accurately
Q3: Text Handling Effectiveness of the VISA4D Tool	The tool handled text inputs very effectively. The tool handled text inputs effectively. The tool handled text inputs neither effectively nor ineffectively. The tool handled text inputs ineffectively. The tool handled text inputs very ineffectively.
Q4: Voice Handling Effectiveness of the VISA4D Tool	The tool handled voice inputs very effectively. The tool handled voice inputs effectively. The tool handled voice inputs neither effectively nor ineffectively. The tool handled voice inputs ineffectively. The tool handled voice inputs very ineffectively.
Q5: VISA4D's Response Time of Commands	The tool's response time to commands was always satisfactory. The tool's response time to commands was usually satisfactory. The tool's response time to commands was sometimes satisfactory and sometimes unsatisfactory. The tool's response time to commands was usually unsatisfactory. The tool's response time to commands was always unsatisfactory.
Q6: Errors in Command Execution by the VISA4D Tool	The tool executed commands without any errors. The tool rarely executed commands with errors. The tool sometimes executed commands with errors. The tool often executed commands with errors. The tool always executed commands with errors.
Q7: User-Friendliness of the VISA4D Tool's Interface	The tool's interface is very user-friendly. The tool's interface is generally user-friendly. The tool's interface is neither user-friendly nor user-unfriendly. The tool's interface is somewhat user-unfriendly. The tool's interface is very user-unfriendly.
Q8: Ease of Learning to Use the VISA4D Tool	Learning to use the tool was very easy. Learning to use the tool was easy. Learning to use the tool was neither easy nor difficult.

Learning to use the tool was difficult.

Learning to use the tool was very difficult.

Q9: What do you like the most about VISA4D?

Q10: What challenges did you face while using VISA4D?

Q11: Do you have any general comments or feedback about VISA4D?

Q12: Are there any additional features you would like to see in VISA4D?

Q13: What improvements would you suggest for VISA4D?

Input: Update the foundation pouring task to complete

SpaCy Processing

Tokenization: [Update][the][foundation][pouring][task][to][complete]

Part of speech: update(verb) foundation(noun) pouring(verb) complete(adj)

Sentence Transformer

Converting to semantic vector representation

Dimension: 384, Optimized for similarity comparison

Construction Pattern Matching

Detected: "foundation pouring" matches construction task pattern

Task Type: Foundation Work, Action: Status Update

Figure A1. Example of an NLP processing pipeline

Input: "Can you create a new task for windows installation?"

Text Processing

Create a new task for windows installation.

Removed stopwords: 'can', 'you', 'a', 'for'

Intent Classification

Create_task (confidence: 0.95)

High confidence due to clear 'create' keyword

Entity Extraction

```
{  
  "task_name": "Windows Installation",  
  "confidence": "0.95"  
}
```

Matched 'installation_pattern' in construction patterns

Figure A2. Example of creating a task

Input: "Update the start date of railing to Sep 24, 2025?"

Text Processing

Update the start date of railing to September 24th, 2025

Removed stopwords: 'the', 'of', 'to'

Intent Classification

Update_date (confidence: 0.93)

Matched date update pattern

Entity Extraction

```
{
  "task_name": "Railing",
  "date": "September 24, 2025",
  "confidence": "0.95"
}
```

Date parsed from Sep 24th

Figure A3. Example of updating date

Input: "Modify the status for painting to in progress?"

Text Processing

Modify status painting progress

Removed stopwords: 'the', 'for', 'to', 'in'

Intent Classification

Update_date (confidence: 0.93)

Recognized status update intent

Entity Extraction

```
{
  "task_name": "Painting",
  "status": "in progress",
  "confidence": "0.94"
}
```

Status matched from status_mapping dictionary

Figure A 4. Example of modifying the status

POST/api/timeliner/task

Create new task

Request:

```
{
  "taskName": "Foundation Work",
  "taskType": "construct",
  "plannedStartDate": "2025-02-15",
  "plannedEndDate": "2025-03-15"
}
```

Response:

```
{
  "success": true
}
```

(a)

PUT/api/timeliner/task/update

Update existing task

Request:

```
{
  "taskName": "Foundation Work",
  "updates": {
    "newName": "Update Foundation Work",
    "newStartDate": "2025-03-20",
    "newEndDate": "2025-03-20",
    "newStatus": "In Progress"
  }
}
```

Response:

```
{
  "success": true
}
```

(b)

DELETE/api/timeliner/task/delete/{taskName}

Delete task

Response:

```
{
  "success": true
}
```

(c)

Figure A5. The three core REST API endpoints for handling tasks (a) POST, (b) PUT, and (c) DELETE

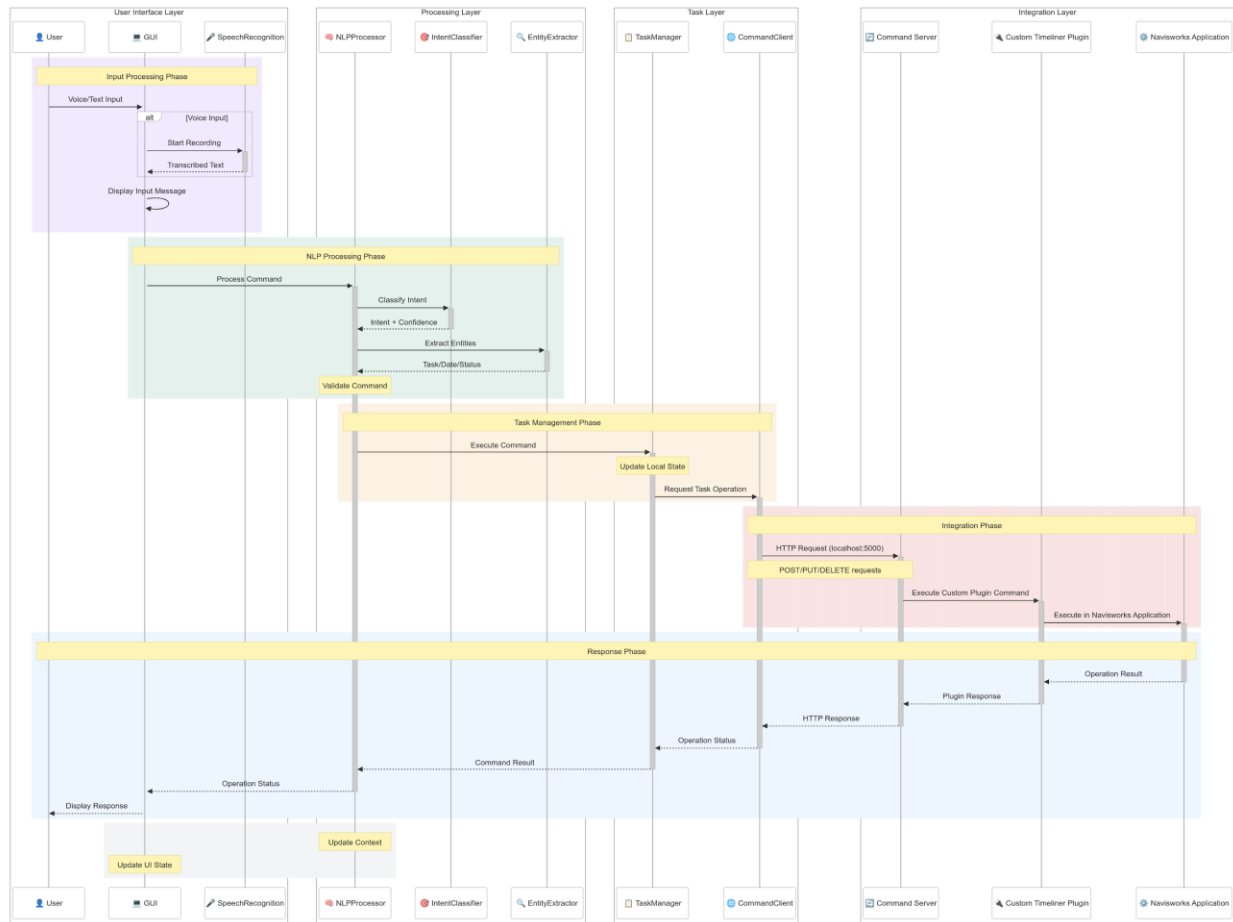


Figure A6. VISA4D communication data flow

APPENDIX B



Civil Engineering
75 Chancellors Circle
University of Manitoba (Fort Garry
campus)
Winnipeg, MB, R3T 5V6, Canada
T: 204 474 9220
F: 204 474 7513
civileng@umanitoba.ca

Evaluation of the VISA4D tool for Updating and Maintenance of Construction Schedules

Consent Form

Principal Investigator:

Assistant Professor: Dr. Gursans Guven Isin, PhD
Department of Civil Engineering
Price Faculty of Engineering
University of Manitoba

Co-Investigator:

Master of Science Student: Lilly Jaff, EIT
Department of Civil Engineering
Price Faculty of Engineering
University of Manitoba

Co-Investigator:

Undergraduate Student: Sahej, Garg
Department of Electrical and Computer Engineering
Price Faculty of Engineering
University of Manitoba

Conflict of Interest:

None of the researchers has any conflicts of interest in this study/research.

This consent form, a copy of which has been given to you, is only part of the process of informed consent process. If you want more details about something mentioned here, or information not included here, feel free to ask any of the people named above. Please take the time to read this document and any accompanying information carefully. It is very important that you understand:

- what is being asked of you,
- what the risks and benefits of participation are, and
- how the information you provide will be used and stored.

Purpose of the Study:

We are doing this research for a Master of Science thesis and a journal paper under the supervision of Dr. Gursans Guven Isin. This research study is about updating and maintaining construction schedules using a Natural Language Processing (NLP)-based tool named **Voice-Integrated Scheduling Assistant for 4-Dimensional (4D) Building Information Model (VISA4D)**. We are hoping to learn about VISA4D's functionality, ease

of use, and users' perspective regarding the tool. Your participation will help improve the tool's effectiveness and usability. The results will be used as part of a Master of Science thesis and academic publications, such as journal articles and/or conference papers.

Study Procedures:

To access VISA4D, our team will provide you with a link to a website that contains instructions and tutorials on using the tool. After successfully finishing all the tutorials, you will be directed to the VISA4D interface, where you can update and change the construction schedule using voice or text commands. After using the tool, we will ask you to complete an anonymous 5-minute survey to provide feedback. We will email you a link and a QR code to access the survey.

Who Can Participate:

You can participate if you are a:

- Undergraduate Civil Engineering Student at the University of Manitoba
- Graduate Civil Engineering Students at the University of Manitoba.

Voluntary Participation:

Your participation is entirely voluntary.

Confidentiality:

This study involves the collection of anonymous data, meaning no personal identifiers will be collected.

Study Risks:

This survey is minimal risk, meaning that the risks are no greater than what the participant might encounter in their daily life.

Study Benefits:

There are no direct benefits or remuneration to you by participating in this study, but you will learn about the study regarding improving and streamlining updates and maintenance in construction schedules using the 4D BIM technology and NLP.

Your feedback will contribute to the refinement of this tool, which can potentially improve the schedule updating and maintenance processes in construction projects.

Storage and Use of Data:

The data will be stored in the following manners:

Storage – Physical Documents	No physical documents will be created or stored as part of this study.
Storage and Backup - Virtual	All consent forms, email communications, and survey forms will be stored on the University of Manitoba's OneDrive platform, which is a secure, UM-approved storage solution.

- I may freely leave the research study activities at any time.
- I do not waive my legal rights by participating in the study.

Please review the following statements. For each statement, please check if you agree or do not agree.

I am an undergraduate or graduate student in the Department of Civil Engineering, Price Faculty of Civil Engineering, University of Manitoba.	☐ YES ☐ NO
---	--

I agree to participate in this study	☐ YES ☐ NO
--------------------------------------	--

I agree that the information I provided to the research team can be used for future research purposes by members of this study's research team and/or their students.	☐ YES ☐ NO
---	--

I agree to be contacted by a member of this research team for future phases related to this research study. If yes, please provide an email address: _____ By agreeing to be contacted, you are not required to participate in any future phases. If you wish to withdraw your name from this list, please contact Lilly Jaff at Jaffl@myumanitoba.ca	☐ YES ☐ NO
--	--

For online studies: If I choose to withdraw partway through the study, I agree that the research team may still include the responses I have completed in their analysis.	☐ YES ☐ NO
---	--

Contact Information:

If you have any questions or concerns about this study, please contact Lilly Jaff at jaffl@myumanitoba.ca

_____	_____	_____
Name of Participant	Participant's Signature	Date

Summary of Results:

Please provide an email address below if you would like to be sent a summary of the study results.

Email address: _____

A summary of the results will be emailed to you by Dec 31st, 2025