buildings

MDPI

Article

BIM-Based Automated Code Compliance Checking System in Malaysian Fire Safety Regulations: A User-Friendly Approach

Aimi Sara Ismail [1], Kherun Nita Ali [1,*], Noorminshah A. Iahad [2], Mukhtar A. Kassem [1,*] and Najib Taher Al-Ashwal [3]

[1] Department of Quantity Surveying, Faculty of Built Environment & Surveying, Universiti Teknologi Malaysia, Skudai 81310, Johor, Malaysia; aimisara@utm.my
[2] Department of Information Systems, Faculty of Management, Universiti Teknologi Malaysia, Skudai 81310, Johor, Malaysia; minshah@utm.my
[3] Department of Architecture, Faculty of Built Environment and Surveying, Universiti Teknologi Malaysia, Skudai 81310, Johor, Malaysia; najib.alashwal@utm.my
* Correspondence: b-kherun@utm.my (K.N.A.); mukhtarkas@gmail.com (M.A.K.)

Abstract: Developing automated code compliance checking systems is becoming increasingly complex—to the extent of challenging the implementation of these systems. This paper addresses the need to develop an automated system that prioritises user accessibility. Accordingly, the study aims to develop a system through a semi-automated rule translation process and the utilisation of BIM models in native file format. A total of 256 fire safety clauses in Malaysian regulations were translated through logic-based approaches (classification technique, decomposition through semantic mark-up method, and interview method), which further assisted in identifying the necessary BIM properties. A visual programming language was then utilised to demonstrate the proof-of-concept prototype. The classification technique and semantic mark-up method were established and structured in this study by developing a framework and flowchart to provide specific guidelines for formalizing the clauses. The semi-automated translation process encouraged the participation of relevant regulatory experts and provided more user accessibility compared to existing studies. This study also offered more practicality for designers to employ the system by utilizing native BIM model data representation. High mean scores ranging from 4.09 to 4.96 were obtained for the validation process, which affirmed the feasibility of the BIM-based Automated System for Malaysian Code Compliance Checking (BIMSMACC) to assist designers.

Keywords: building information modelling; compliance checking; semantics; knowledge representation; visual programming language

check for updates

Citation: Ismail, A.S.; Ali, K.N.; Iahad, N.A.; Kassem, M.A.; Al-Ashwal, N.T. BIM-Based Automated Code Compliance Checking System in Malaysian Fire Safety Regulations: A User-Friendly Approach. *Buildings* **2023**, *13*, 1404. https://doi.org/10.3390/buildings13061404

Academic Editor: Maziar Yazdani

Received: 9 April 2023
Revised: 22 May 2023
Accepted: 27 May 2023
Published: 29 May 2023

1. Introduction

The Architecture, Engineering, and Construction (AEC) industry has undergone a remarkable evolution from dependence on two-dimensional (2D) Computer-Aided Design (CAD) building plans to the digitalisation and automation of design, construction, and operational processes through Building Information Modelling (BIM). Advances in BIM technology have stimulated a wide range of approaches to automating the code compliance checking process. Conventionally, designers and local authorities rely on 2D building plans to manually check the compliance of these plans with relevant building regulations. As the complexity of BIM models has increased, this procedure has become more time-consuming and could result in ambiguity and inconsistency in design assessments [1]. The highly subjective nature of building regulations could further lead to inefficiency in the manual checking process [2]. The automation of this process involves computer-programmed analysis of the building objects and attributes based on a set of regulations to produce a compliance report. An automated code compliance checking system promotes greater accuracy, consistency, and comprehensiveness in the design process compared to the manual checking procedure.

The essential components of an automated code compliance checking system are the rules and building objects [3]. The building regulations should first be interpreted, followed by the identification of necessary building objects for the checking process. The relationship between these components should be determined to demonstrate the checking process. Given the complex nature of building regulations, a wide range of studies has explored approaches to interpreting the reasoning behind them. Early studies in the 1990s encouraged the development of expert systems in the CAD environment [4–7]. The emergence of BIM technology in the early 2000s propelled the development of automated systems to support the BIM-based environment. The CORENET e-PlanCheck system in Singapore [8], Solibri Model Checker (SMC) in Finland [9], DesignCheck in Australia [10], SMARTcodes and spatial program validation in the United States [3,11], and Statsbygg's design checking in Norway [12] are among the prominent systems of the early 2000s. These automated systems were built through hard-coding techniques, also known as the 'Black Box' method. Despite its ability to eliminate errors during the rule execution process, this method hinders the involvement and accessibility of users to assess, modify, and extend the rules in the database [3,13–15].

Consequently, the application of a 'White Box' method has become prevalent in recent studies. This method provides greater transparency of the rule interpretation and execution process for users [13,15]. Examples of the White Box method are the semantic mark-up RASE (Requirement, Applicability, Selection, and Exception) methodology [16], natural language processing (NLP) techniques [17], visual programming language (VPL) techniques [18], and an ontological approach through semantic web technologies [19]. Despite the wide variety of approaches, challenges remain in the actual implementation of the automated system. Although the NLP and ontological approaches have a high level of success in translating complex rules, they have been argued to be impractical for AEC domain experts because of their complexity [18,20]. Parallel to the development of these approaches, logic-based approaches have been explored to extract the semantics of rules without requiring any programming skills [15]. However, they include a preprocessing rule interpretation procedure and cannot directly transform the rules into computer representation. Alternatively, the VPL technique allows direct integration and greater accessibility for users to review and modify existing rules, as well as create new rules [13,21]. Despite the strength of VPL, the logic-based approaches remain relevant to avoid logical errors [22].

For the identification of necessary building objects, existing studies favour the representation in BIM models of the Industry Foundation Classes (IFC) format, to promote better data interoperability. However, the modelling requirements for IFC data are known to be complex and require proficiency in the EXPRESS language [23,24]. This procedure tends to misclassify some IFC data, which could lead to data losses and errors [23,25–27].

The challenges to the automated code compliance checking field of study thus hinder the actual implementation in the real world. The challenges and limitations presented in past studies revealed the research gap in this field, in which the practicality for users were not prioritised during the development of an automated system [28]. In conjunction with this, AEC experts should be involved in developing the rule database, rather than solely software developers [15]. The separation of the rule interpretation process and the rule execution process could also provide greater accessibility, enabling users to maintain the rules during their use of the system. Using BIM models in native file format could also increase users' accessibility. Thus, this study aims to develop an automated code compliance checking system through a user-friendly approach that includes a semi-automated rule translation process and the utilisation of BIM models in native file format, specifically the Revit file format.

The remainder of the study is organised as follows. In the next section, the background literature is reviewed to identify the strengths and limitations of the existing approaches and suggest a way forward. A framework containing the methods in developing the system is presented in Section 3. Section 4 demonstrates the implementation and validation of the framework in a case study, and Section 5 presents the discussion of the results. The conclusion is set out in the final section.

2. Literature Review

Over recent decades, there has been much debate over representing the rules and building objects as deduced from either the Black Box or the White Box method [3,15]. Although the Black Box method was viewed as unsuitable for AEC experts due to its limited access by users, this method was proven to be successful according to actual implementations reported in earlier studies. However, the advantages of the White Box method make it more prevalent nowadays. Thus, it is important to review both methods to identify their strengths and limitations.

2.1. Existing Applications

This paper highlights several applications that remain relevant: CORENET BIM e-Submission, BIM Assure, Jotne Express Data Manager (EDM), Solibri Model Checker (SMC), and Daima. These applications were all built through hard-coding techniques. The CORENET project in Singapore is considered the first successful initiative [8]. It is a web-based service that initially focused on 2D drawings (CORENET BP-Expert) but later shifted towards 3D model checking (CORENET e-Plan Check) and is now known as CORENET BIM e-Submission [29]. Similar to the development of CORENET e-Plan Check, DesignCheck was developed in Australia [10] to support 2D checking. Although it was stated that the future development of DesignCheck would include 3D model checking, the initiative was discontinued [9].

Several software applications have emerged to specifically analyse and check building models on a BIM platform. The most established software application is SMC, a stand-alone application that contains some 300 preformatted rules related to egress, space management, and accessibility [2,15]. Users are only required to select the rulesets, modify the parameters, and simply click the ‘CHECK’ button to execute the automated checking process. SMC was tested in the HITOS project in Norway [3] and is currently being employed for BIM model validation in the United States [30]. Other emerging applications are BIM Assure, released in 2016 [31], and Daima, released in 2019 [32]; both are plugins for Revit, in which BIM Assure has the advantage of supporting a cloud-based environment.

The existing applications have one major shortcoming, the users’ flexibility to create new rules is limited. In CORENET BIM e-Submission, the Singaporean building regulations were represented through an object-based approach on the independent platform FORNAX [3,8]. FORNAX is a C++ object library where the rules are hard coded into the system and matched with the IFC objects. For DesignCheck, Jotne EDM offered a backend rule engine to represent and manage the rules; however, the modification of the rules could only be conducted by software developers [3,15,33]. This limitation also exists in SMC, where users are allowed only to customise the parameters of the preformatted rules [2,34]; the creation of new rules can only be performed upon request, under a fee-based subscription [2,15,35]. In BIM Assure, the creation of new rules is limited to non-geometrical parameters [36], while in Daima the creation of new rules is again the domain of the software developer [37]. It was argued that although the hard-coding technique in the earliest applications was proven to be reliable, the maintenance and extension of rules were difficult to execute [14]. For this reason, the transparency and flexibility for users to assess the rule database were considered an advantage in the development of newer automated systems [13,15]. Past studies have suggested the representation of rules in a logical approach to be more accessible for users [3,35].

2.2. Logic-Based Approach

In general, the logic-based approach consists of the decision table, semantic, and knowledge base methods. The decision table method was the earliest effort in rule interpretation, where steel design specifications were documented in the form of parametric tables [38], comprising logical conditions in the rules and the appropriate actions to be executed. This method was later adopted and modified to check the designs of the building envelope [39]. It is considered that decision tables can concisely represent the rules.

Parallel to the adoption of decision tables, the mark-up concept was introduced for the interpretation of building codes in North America [3]. A tool known as SMARTcodes builder was developed to mark up the rules in different colours, which resembled an electronic version of highlighting pens [40]. Despite its initial discontinuance due to insufficient funding, the markup concept was further explored and formally introduced as the RASE (Requirement, Applicability, Selection, and Exception) methodology [41]. The RASE methodology was tested in selected rules in the Norwegian accessibility standards, Dubai building regulations, and United States Court design guidance documents. The selected rules were marked according to four concepts: Requirement, Applicability, Selection, and Exception. This simple and well-defined process motivated other researchers to adopt this technique for rule representation, in Code for Sustainable Homes (CSH), Building Research Establishment Environmental Assessment Method (BREAAM) [42], Life Safety Codes (LSC) in Australia [33], and Turkish regulations [43] as well as to create new rules in SMC [44].

Another prevailing technique in interpreting building regulations on a logical basis is the knowledge base approach, which emphasises the separation of the rule database from the rule execution process. ACABIM, an automated compliance audit tool, is an example of a system that applies the knowledge base approach; it consists of a building compliance model, regulatory knowledge model (RKM), and compliant design procedure [45]. The RKM is the rule database for the system, storing open standard representations of the building codes. BIM Rule Language was also developed to support the complexity of the semantics in the building codes [46]. Other examples of the knowledge base approach are the conceptual graph, which represents the rules with nodes and arrows [47], and first-order logic (FOL), which is also known as predicate logic [11,48–50]. For conceptual graphs, it was argued that the representation process was time-consuming and costly [51], while FOL was unable to accommodate the various types of logic and information in the building regulations [52].

2.3. Natural Language Processing (NLP) Technique

The logic-based approaches were considered a preprocessing procedure requiring an encoding process to integrate the rules into the system [15]. Other studies preferred the direct integration of rule representation into the proposed automated systems, which could be achieved through NLP techniques. NLP was prevalent in the chain of studies conducted by Zhang and others since 2011 [53], and Zhou and others since 2014 [54]. In both, the application of information extraction (IE) and information transformation (ITr) was combined with logic-based reasoning algorithms to represent the rules and execute the checking process [20,55]. In one of the studies, the information in the rules was extracted through text pre-processing, syntactic parsing, and semantic analysis [17]. The reasoning algorithm for the rules was then developed through FOL [56]. The other chain of studies demonstrated the application of Text Classification before implementing the IE methodology [55]. These studies then motivated others to explore the NLP technique in proposing their automated systems. For instance, the Korean Building Act was translated through a deep-learning-based predicate-argument-structure extraction process to improve the computer representation of rules, known as KBimCode [57,58]. Another example is the development of machine learning algorithms to increase the accuracy of tagging on the rules [59].

The NLP technique was proven to be reliable based on the high percentages of recall and precision obtained from performance testing [17,55]. However, it was argued that the case studies in the tests consisted of straightforward and well-defined regulations [15]. Furthermore, it was highlighted that high proficiency in the domain of related building regulations was required to successfully implement the NLP technique [20]. Inputs from relevant AEC experts were still required to define the semantics and ontologies of rules and BIM objects in the application of machine-learning-based and deep-learning-based approaches [55,58]. This led to a huge amount of work in mapping the rules with the BIM objects, which vary according to the building regulations and IFC representations [20,58].

2.4. Visual Programming Language (VPL) Technique

Assuming limited programming skills among AEC professionals, the VPL approach was introduced to represent the mapping between the rules and BIM data [28]. This approach represents data in a diagrammatic manner consisting of nodes, each of which contains receive statements and send statements [60]. One of the earliest studies demonstrated the development of domain-specific VPL programs known as Visual Code Checking Language (VCCL) programs [28]. The VCCL programs were further refined through a nesting approach to reduce their complexity [18]; an additional visual language known as VQL4BIM was also developed to support the data retrieval of IFC models [61]. Similar to the approach of VCCL programs, a visual representation of the Korean Building Act was developed known as KBim Visual Language (KBVL) [50,62]. The Korean Building Act had previously been translated through FOL [63] but was further represented through KBVL to provide more visibility for AEC professionals. Other studies implemented the VPL approach for the overall thermal transfer value (OTTV) checking process [64] and for the representation of specific legal document open standards [13,65].

2.5. Ontological Approach Based on Semantic Web Technologies

Semantic web technology was promoted as a means to improve data interoperability in the AEC industry as it could centralise the data on a single platform and link it to various applications [24]. The earliest examples of this approach [39,66] in the automation of the code compliance checking process represented the rules in XML format. However, the XML format was considered inadequate as it could not process and combine the information [67]. A more advanced semantic web technology was then explored through the creation of a framework for a formal ontological approach [68]. The regulations were formalised into SPARQL queries and represented in OWL-Lite ontology, while the building model was represented in ifcXML [69]. The components of each were matched through a reasoning model and integrated into a prototype called the C3R system. Although accessibility was the main strength of this system, the spatial relationships and properties in the IFC model were disregarded [18]. The SPARQL query language has also been applied in more recent studies, such as building environmental compliance checking [70] and underground utility compliance checking [71]. Although these studies have proved the workability of this approach, they stressed that the selected regulations were confined to the easier ones.

Another common approach in this field of study was the application of a dedicated rule language such as the Semantic Web Rule Language (SWRL) and N3Logic. Past studies employed the SWRL in Protégé, a free open-source platform. In [72], CQIEOntology was proposed as a meta-model for construction quality inspection and evaluation, which divided the elements into nine classes: Inspection-Object, Inspection-Task, Inspection-Item-Checking-Action, Evaluation-Task, Evaluation-Criteria, Evaluation-Result, Checking-Result, Inspection-Report, and Regulation-Constraint. The application of SWRL has also been seen in other studies [19,73] for an ontology-based job hazard analysis and visualisation system.

2.6. The Way Forward

Based on the limitations and challenges of the existing applications and approaches, this study highlights the requirements for a successful automated checking system as follows:

- To cater for the involvement of AEC and regulatory experts in interpreting and maintaining the rules.
- To provide an independent database management system for rule maintenance.
- To provide accessibility for users to modify existing rules and create new rules.

- To prioritise users in employing the system.

In general, despite the workability of the Black Box method in existing applications, this approach cannot cater for the involvement of AEC domain experts in the rule representation process [3,13,14]. The application of logic-based approaches in past studies, consisting of the decision table method, semantic mark-up RASE methodology, and knowledge base approach, proved that AEC experts could employ these approaches without any programming skills. Based on these three approaches, the semantic mark-up RASE methodology was the most promising method in the rule translation process as a substantial number of studies [42–44,74,75] had tested it in their respective regulations. Alternatively, NLP, VPL, and ontological approaches have the leverage to directly integrate the rules into the system. However, the complexity of the NLP techniques and ontological approach makes them impractical for AEC experts [18,20]. On the other hand, the VPL approach could allow for the involvement of AEC experts and provide accessibility for users to review and modify the rules [13,21]. Combining the logic-based approach with the VPL approach could avoid logical errors [22].

Existing studies have preferred the representation of building models in IFC data models such as the FORNAX objects [8], ifcXML [69], and ifcOWL [76]. The representation of IFC data in BIM-based code compliance checking systems was also recognised in the United States [30] and Norway [77]. However, studies conducted by [25,26] claimed that the misclassification of IFC data was a common problem, failing in BIM data exchange. The potential for data loss also posed a problem in the application of the IFC data model [23,27]. The preparation of an IFC data schema was complicated [78]: first, the data representation process varied among BIM software applications; second, the process involved a tremendous amount of effort; third, only users with knowledge of the EXPRESS language were able to extend the IFC schema [24]. The recent development of CORENET BIM e-Submission allows BIM submission in native file format, specifically in Revit and ArchiCAD, to accommodate users [79]. Thus, the representation of BIM models in native file format is the way forward in developing an automated system.

3. Methods

Based on the knowledge gaps, this paper attempts to answer two key research questions: (1) What is a viable approach to accommodate the involvement of relevant experts in interpreting building regulations? (2) What is a practical approach for users in integrating and maintaining the rules in a BIM-based environment? In consideration with the research questions, the aim of this study is to develop an automated BIM-based code compliance checking system through a user-friendly approach. To accomplish this aim, this study aims to achieve the following objectives:

- To structure the representation of rules and BIM properties for an automated code compliance checking system through a user-friendly approach.
- To develop and validate a prototype of a BIM-based Automated System for Malaysian Code Compliance Checking (BIMSMACC).

A critical realist stance was adopted to critique the existing phenomena and convert them into change or improvement. This stance is aligned with design science research (DSR), which aims to improve the existing knowledge by creating artefacts [80]. The DSR methodology was applied in this study to address the research questions through a problem-based approach [81]. Abductive reasoning was applied to draw the current problems and proposed approaches from the literature and establish suggestions for improvement. Inductive reasoning was then applied to construct the artefacts through a qualitative approach. The artefacts, in the form of a framework and prototype, were demonstrated and validated. Iteration occurred between the stages to refine the artefacts.

Figure 1 illustrates the framework in developing the automated BIM-based code compliance checking system for fire safety regulations. The framework has four components: rule interpretation, building model preparation, rule execution, and rule reporting. These components are similar to the development stages proposed in a previous study [3]. In the rule interpretation stage, the applicable fire safety regulations in Malaysia are formalised through a logic-based approach. This is followed by the building model preparation stage to identify the necessary BIM properties. The rule execution and reporting stage then simulates the code compliance checking process, where experts are involved in validating the system.

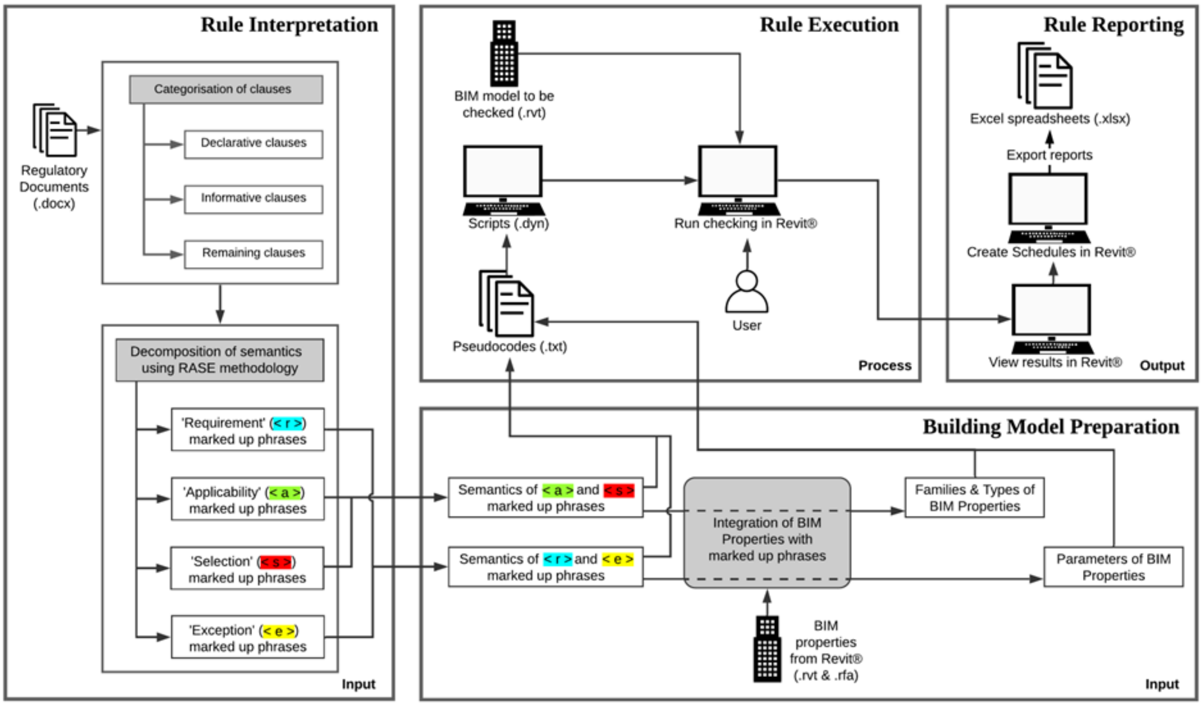


Figure 1. Proposed framework for an automated code compliance checking system.

A total of 256 clauses from Part VII (Fire Requirements) and Part VIII (Fire Alarms, Fire Detection, Fire Extinguishment, and Fire Fighting Access) in the Selangor Uniform Building (Amendment) (No. 2) By-Laws 2012 were selected for the rule interpretation. The authors formalised the building regulations through a classification technique and semantic mark-up RASE methodology. The classification technique was employed to maintain the integrity and structure of the building regulations. Generally, previous studies introduced different classification techniques: (1) the categorisation of clauses into declarative, informative, or remaining clauses [82]; (2) the classification of clauses into Class 1, Class 2, Class 3, or Class 4 [83]; (3) the classification of clauses into Content, Provisory, Dependent, or Ambiguous concepts [84]. The first classification technique [85] was adopted as it includes provision for the clauses that are unsuitable for the automated code compliance checking process.

The semantic mark-up RASE methodology was then applied for declarative and informative clauses. This process decomposed the semantics of building regulations according to four operators: Requirement (<r>), Applicability (<a>), Selection (<s>), and Exception (<e>). The definition of each operator will be discussed in Section 4. This approach was adopted in this paper and compared with other logic-based approaches such as decision tables and knowledge-based approach. Based on these three approaches, the semantic mark-up RASE methodology was the most promising method in the rule translation process as a substantial number of studies [42–44,74,75] had tested it in their respective regulations.

Interview sessions with three fire safety experts were also conducted to assist in interpreting the informative clauses. These interview sessions encouraged the involvement of AEC professionals in interpreting the clauses, as suggested in previous studies [3,15]. In selecting the experts, purposeful sampling was conducted in which the samples were selected based on the knowledge and experience of relevant experts on fire safety design. The first expert was a Branch Chief of the Fire Safety Certification Branch, Fire and Rescue Department of Malaysia (JBPM) with an experience span of 33 years. The other two experts were fire engineers which are abreast with the emerging technologies and performance-based design related to fire safety in buildings.

Next, in the building model preparation stage, the authors identified the necessary BIM properties in native file format, specifically in Revit. A 17-storey institutional building BIM model was created in Revit for this purpose. Revit is popular among designers as it has a shallow learning curve and a user-friendly interface [1,86,87]. In Malaysia, Revit has been the preferred BIM authoring software application since 2013, and the official platform for the Public Works Department (PWD) to create a library of families [88,89]. An organisation under the Construction Industry Development Board (CIDB), the myBIM centre, also uses Revit for their BIM training [90]; the centre currently offers BIM training to AEC professionals to encourage BIM adoption in Malaysia. The identification of necessary BIM properties in the framework is based on the families and parameters in Revit. The necessary families are derived from <a> and <e> marked-up phrases while the parameters are derived from <r> and <e> marked-up phrases. The BIM properties act as input for the rule execution stage.

The rule execution stage then simulated the code compliance checking process. The authors developed pseudocodes as an initial step in establishing the relationship between the formalised rules and BIM properties. Past studies recommended the application of pseudocodes to support the encoding process and functionality of the automated system [3,15,91]. A portion of the pseudocodes were then encoded through Dynamo, a visual programming tool in Revit, to develop a proof-of-concept prototype. The checking process and compliance results could then be run directly in Revit.

As the validation process is a major contribution of DSR [82], thirteen experts were involved in validating the artefacts, to determine their completeness, accuracy, usability, and reliability. The validation process was conducted through interview sessions guided by structured questionnaires to gain feedback and constructive comments from the experts. Five experts—one representative from the fire authority, three fire engineers, and one architect—validated the framework. For the prototype, eight experts (three architects, two structural engineers, one mechanical engineer, and two civil engineers) were involved in the validation process. The experts were chosen for their qualifications, experience, and knowledge of the subject matter. In validating the elements in the system, several samples of formalised rules and BIM properties were presented to the experts. For the validation of prototype, its features and functions were first demonstrated to the experts. Next, the experts were invited to test the prototype through the Dynamo Player in Revit. Subsequently, in both validation processes, the experts provided feedback and filled out the questionnaires. Mean scores were calculated to analyse their answers.

4. Case Study (Implementation and Validation)

This section demonstrates the implementation of the proposed framework in developing a BIM-based Automated System for Malaysian Code Compliance Checking (BIMS-MACC). The study formalised a total of 256 fire safety clauses; identified the necessary BIM properties; developed the pseudocodes; and lastly, created a proof-of-concept prototype.

4.1. Interpretation of Fire Safety Clauses

There are three stages in formalising the fire safety clauses: the categorisation process, the decomposition process, and the interpretation process through interview sessions.

4.1.1. Categorisation of Fire Safety Clauses

In formalizing the categorisation process, Figure 2 was developed, where the authors initially identified the clauses that contain references to other clauses or regulations. This step was necessary due to the high occurrence of the interrelationship between clauses and dependence on the Schedules stipulated in the Selangor Uniform Building (Amendment) (No. 2) By-Laws 2012 as well as other regulations from the Malaysian Standards, British Standards, Australian Standards, and International Electronical Commission Standards. Following this procedure, all clauses, whether marked or not, were analysed in the first filter categorisation.

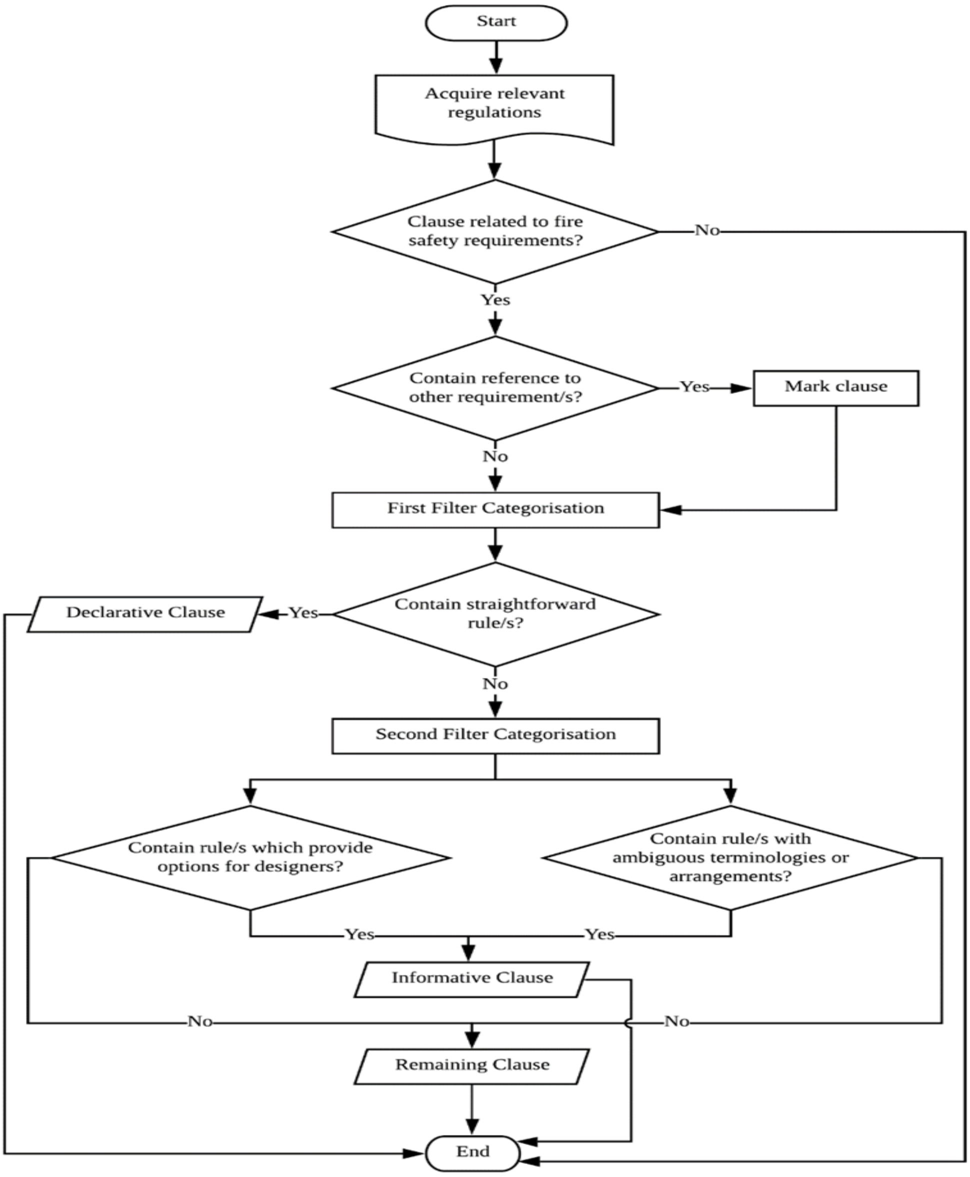


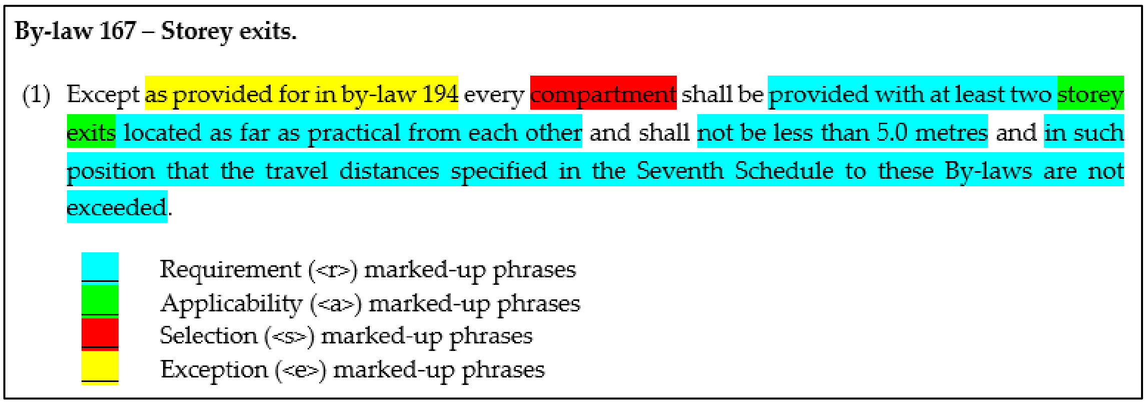
Figure 2. Categorisation process.

As presented in Figure 1, the first filter categorisation identified clauses that contain straightforward rules such as geometrical or definite rules. These were extracted and categorised as declarative clauses. Other clauses were then channelled to the second filter categorisation to identify the informative clauses. Clauses that contain ambiguous terminology or arrangements—for instance, "where appropriate", "suitable", "adequate", "as far as the practical", or "furthermost point of the floor"—were grouped as informative clauses. In other instances, clauses that stated the requirement for special approval from the fire authority, which could lead to ambiguity, were also categorised as informative. Based on Figure 1, Informative clauses also represented any clauses that provide options for designers, such as "either", "where practical", and "where impractical". Clauses that were not suitable for the first and second filters were categorised as the remaining clauses.

Through this categorisation process, a total of 166 declarative clauses, 53 informative clauses, and 37 remaining clauses were identified in the Selangor Uniform Building (Amendment) (No. 2) By-Laws 2012.

4.1.2. Decomposition of Semantics in Fire Safety Clauses

The declarative clauses were initially decomposed through the semantic markup RASE methodology, followed by the informative clauses. The <r> operator signified the rules in a clause and was associated with modal verbs. The modal verb "shall" was the most prevalent in the fire safety clauses to express necessity or prohibition in designing the building. For instance, by-law 167 paragraph 1 (Figure 3) contains the modal verb "shall" to explain the requirements for storey exits. "Will" was also prevalent in other clauses, while "may" was frequently used in informative clauses to imply the design alternatives. The rules could also indicate prohibitions by having the word "no".



By-law 167 – Storey exits.

(1) Except as provided for in by-law 194 every compartment shall be provided with at least two storey exits located as far as practical from each other and shall not be less than 5.0 metres and in such position that the travel distances specified in the Seventh Schedule to these By-laws are not exceeded.

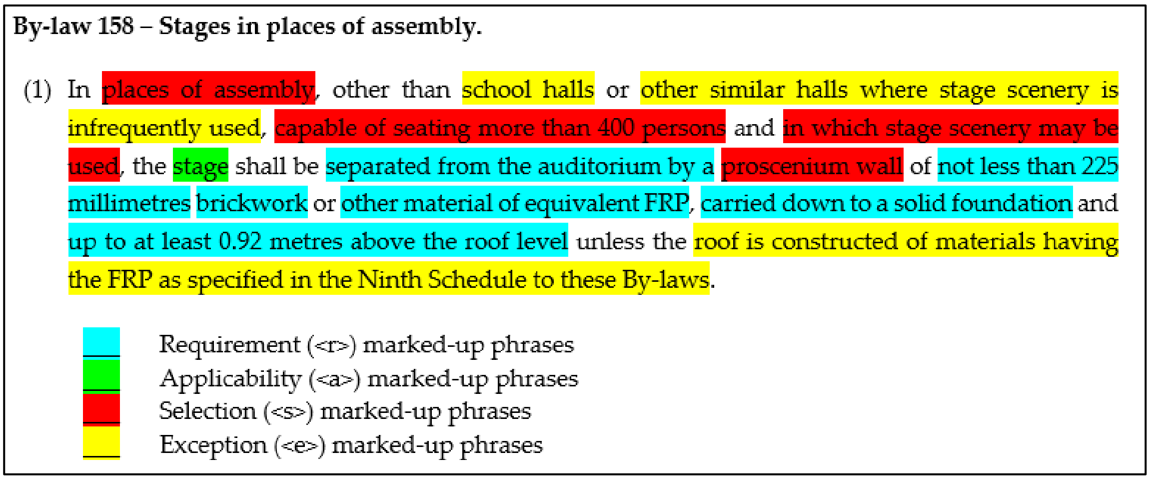
🟦 Requirement (<r>) marked-up phrases
🟩 Applicability (<a>) marked-up phrases
🟥 Selection (<s>) marked-up phrases
🟨 Exception (<e>) marked-up phrases

Figure 3. Marked-up phrases in by-law 167 paragraph 1.

The <a> and <s> operators represent the objects in the clauses. The <a> operator was applied to mark the main object in a clause while other objects were marked with <s> operators. In identifying the main object in a clause, imperative sentences that convey instructions or prohibitions were analysed. Subjects of the imperative sentences were considered as the objects in a clause. Due to their complex composition, many clauses contain more than one subject. Hence, the most distinct technique in identifying the main objects was to analyse the title of the by-laws. Since most by-laws contain more than one paragraph, the title represents the theme of the clauses under the same by-law, such as by-law 167 paragraph 1 (Figure 3). In other circumstances, the main object in a clause was identified by referring to other clauses of the same by-law. Objects other than the main object were then marked with the <s> operator.

Phrases were then marked with the <e> operator to signify the exception to the rules in the clause, such as in Figure 3, and to identify the exceptions of <a> and <s> marked-up phrases. By-law 158 paragraph 1 in Figure 4 provides an application of the <e> operator to mark the exceptions of <s> marked-up phrases. Keywords such as "except", "unless", and "other than" were used to determine the <e> marked-up phrases.

The clauses with marked-up phrases were then illustrated in a tabular manner, as shown in Table 1 Each marked-up phrase was ascribed to a topic, comparator, and target value. For <e> marked-up phrases, the topics were determined based on their nature. If the <e> marked-up phrases were exceptions to <r> marked-up phrases, then the topics would be the rules imposed in <r> marked-up phrases. On the other hand, if the <e> marked-up phrases were exceptions to either <a> or <s> marked-up phrases, then the topics were determined based on the topics of <a> or <s> marked-up phrases.



By-law 158 – Stages in places of assembly.

(1) In places of assembly, other than school halls or other similar halls where stage scenery is infrequently used, capable of seating more than 400 persons and in which stage scenery may be used, the stage shall be separated from the auditorium by a proscenium wall of not less than 225 millimetres brickwork or other material of equivalent FRP, carried down to a solid foundation and up to at least 0.92 metres above the roof level unless the roof is constructed of materials having the FRP as specified in the Ninth Schedule to these By-laws.

🟦 Requirement (<r>) marked-up phrases
🟩 Applicability (<a>) marked-up phrases
🟥 Selection (<s>) marked-up phrases
🟨 Exception (<e>) marked-up phrases

Figure 4. Marked-up phrases in by-law 158 paragraph 1.

Table 1. Decomposition of marked-up phrases in by-law 167 paragraph 1.

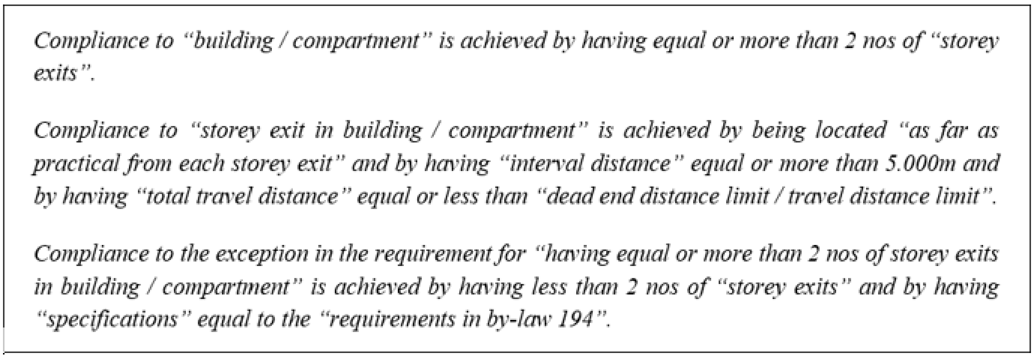
Metric Phase	Type	Object	Property	Comparison	Target	Unit
as provided for in by-law 194	<e>	Less than 2-storey exits in building/compartment	Specifications	Includes	Requirements in by-law 194 [REFER BY-LAW 194]	-
compartment	<s>	Building	Type	Includes	Building/Compartment	-
provided with at least two-storey exits	<r>	Building/Compartment	Storey exits	Equal or more than	2	Nos
storey exits	<a>	Exit	Type	Includes	Storey exit	-
located as far as practical from each other	<r>	Storey exit in building/compartment	Location	Equal	As far as practical from each storey exit [REQUIRE FURTHER INTERPRETATION]	-
not less than 5.0 metres	<r>	Storey exit in building/compartment	Interval distance	Equal or more than	5.000	m
in such a position that the travel distances specified in the Seventh Schedule to these by-laws are not exceeded	<r>	Storey exit in building/compartment	Total travel distance	Equal	Dead end distance limit/Travel distance limit [REFER SEVENTH SCHEDULE]	-

Next, the comparators and target values were identified for each phrase, as shown in Table 2. The "equal", "not equal", "more than", and "less than" comparators were applied for <r> marked-up phrases while the "include" and "exclude" comparators were applied for <a>, <s>, and <e> marked-up phrases. Any marked-up phrases containing references to other clauses or regulations were identified in the table, as shown in Table 1, which was similar to the procedure in the categorisation process. Any topics, comparators, and target values in informative clauses that were ambiguous or provided options for designers were also labelled to acquire further interpretation from relevant fire safety experts.

Table 2. Integration of <a> and <s> marked-up phrases with families in Revit.

Clause	Type	Marked-Up Phrase	Family in Revit
169(2)	<s>	Hospital and nursing home of Purpose Group II (Institutional)	Project information
197(1)	<s>	Buildings in which the topmost floor is more than 18.0 metres above the fire appliance access level	Building elevation
220	<a>	The maximum floor area	Floor plan
197A(2)	<a>	Firefighting access lobbies	Room
148(1)	<a>	Compartment wall	Wall
168(2)	<a>	Staircases	Stairs
185(4)	<a>	Steps	Runs
169(2)	<a>	Landings	Landing
168(3)	<s>	Handrails	Railing (Stairs)
184(1)	<a>	Rows of seats between gangways	Furniture assembly
140(1)	<a>	Access way	Topography (Access way)

The topics, comparators, and target values were organised to restructure the fire safety clauses so as to validate the accuracy of the decomposition process. The restructured clauses were written in prose form, as displayed in Figure 5.



Compliance to "building / compartment" is achieved by having equal or more than 2 nos of "storey exits".

Compliance to "storey exit in building / compartment" is achieved by being located "as far as practical from each storey exit" and by having "interval distance" equal or more than 5.000m and by having "total travel distance" equal or less than "dead end distance limit / travel distance limit".

Compliance to the exception in the requirement for "having equal or more than 2 nos of storey exits in building / compartment" is achieved by having less than 2 nos of "storey exits" and by having "specifications" equal to the "requirements in by-law 194".

Figure 5. Restructured clause of by-law 167 paragraph 1.

4.1.3. Interpretation of Clauses by Fire Safety Experts

A total of 53 informative clauses were further interpreted through interview sessions with a Branch Chief of the Fire Safety Certification Branch, the Fire and Rescue Department of Malaysia and two fire engineers. These interview sessions encouraged the involvement of AEC professionals in interpreting the clauses, as suggested in previous studies [3,15]. Four topics were directed to the fire safety experts: (1) exits and travel paths, (2) building elements, (3) ventilation systems, and (4) active fire protection systems. The fire engineers were responsible for interpreting clauses related to the design considerations, while the fire authority interpreted any clauses stating a requirement for special approval.

The fire engineers stated that determining travel distances without technological tools could be very subjective and prone to error. Thus, they believed that the interpretation of the furthermost point of the floor should not be derived from human judgement but instead from a computer application. Nevertheless, one of the respondents indicated that fire risk analysis is still relevant to determine the optimum travel paths. As stated by the fire engineers, the travel paths and building layout could influence the determination of the most practical distance between storey exits while also referring to other international regulations. Based on the fire safety clauses, the storey exits should be readily accessible at all times. The fire engineers interpreted that the doors should be situated within travel distance limits and unlocked during a fire incident. Thus, the locking mechanism of storey exits could be incorporated into the checking process. The fire safety clauses also mentioned the flexibility of designing horizontal exits in buildings. The fire engineers believed that the lack of clarity in this clause could potentially lead to evacuation problems. In this case, they suggested the use of a performance-based design to determine the availability of horizontal exits.

The informative clauses also provide options for designers in arranging the firestops and flame barriers for structural elements. As interpreted by the fire engineers, the placement of firestops and flame barriers are subject to fire risk analysis but should achieve the minimum fire resistance period requirement of the structural elements. For other design considerations that require special approval from the fire authority, such as protection for lift shaft openings, roof construction, and gangway layout, the Branch Chief officer stated that the fire authority would allow flexibility for designers to produce the appropriate fire safety solutions. Thus, this shows that although the fire authority has superior power in approving building designs, the responsibility and technical competence of designers to conduct a comprehensive compliance check are much higher than that of the fire authority.

For the arrangement of ventilation systems, ambiguities arose in the informative clauses over their design in protected lobbies and atria as well as the use of natural draught smoke vents. For the functional requirements of the pressurisation system in protected lobbies, the fire authority suggested applying Malaysian Standard 1472 (MS 1472). With the emergence of many pressurisation system products, the Branch Chief officer added that the fire authority could adapt to any new system that meets the requirements. Pressurisation systems were also strongly encouraged by the fire engineers in the staircase enclosures below ground floor level. For natural draught smoke vents, the fire engineers explained that these should be designed according to the smoke layer height, the size of the smoke reservoir, and the building's function and geometry. Designers could refer to MS1780 for this purpose. The Branch Chief officer stressed that any closed-design smoke vents should open automatically with the activation of the fire alarm system and be subjected to the fire authority's approval. Ambiguities also occurred in the requirements for designing ventilation ducts. In this case, the fire engineers interpreted that the arrangement of these ducts and their components largely depends on their analysis of the building.

As stipulated in the fire safety clauses, any active fire protection systems must be approved by the fire authority. Thus, as interpreted by the Branch Chief officer, the requirements in informative clauses should be followed based on the suitability of the building. For instance, the provision for additional fire hydrants depends on the building's geometry, type, dimensions, and gradient of accessway. The protection method for the storage of fire hazardous material should also be based on the type of fire hazard in the building spaces. The Branch Chief officer explained that the components of fire protection systems, such as the material of cables and the capacity of the systems, should be suggested by the designers. All respondents believed that every building is unique and the design solution for one building may not apply to another building. They further stated that there is no rule of thumb regarding compliance with informative clauses in the fire safety regulations. The findings from these interview sessions were used as a basis for the target values in the rule execution process. In this case, designers and the fire authority could override the compliance results for informative clauses.

4.2. Identification of BIM Properties

The fire safety clauses were first arranged according to the building elements and spaces in defining the necessary BIM properties. Next, the marked-up phrases for each clause were extracted and integrated with BIM properties. As previously discussed, the <a>

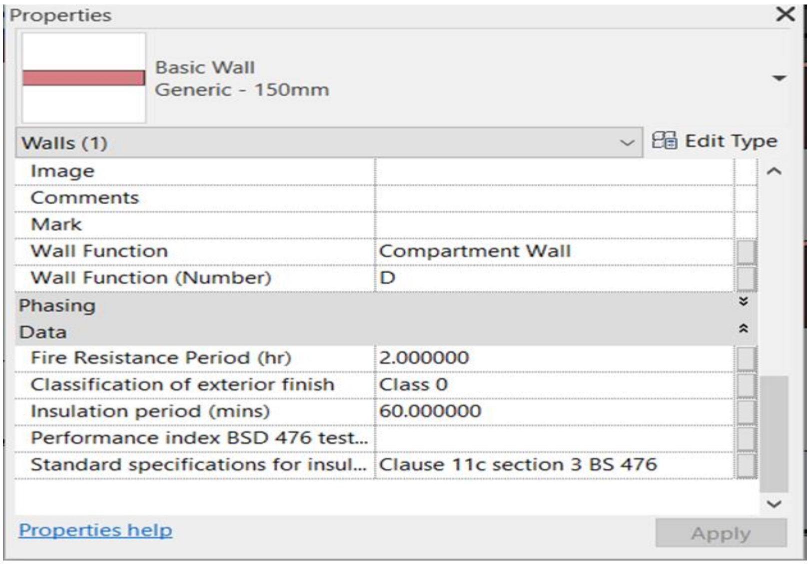
and <s> marked-up phrases represented the objects. Thus, these phrases were integrated with the families in Revit. Table 2 provides examples. A total of 54 families in Revit were identified as necessary for the fire regulation compliance checking process. In ensuring standardisation in naming the families, the OmniClass™ classification system was used, and the numbers and descriptions of OmniClass™ (Table 2) were added as parameters to the families. For example, the fire hose reels were represented by OmniClass Numbers 23–29 25 15 17.

The <r> and <e> marked-up phrases were then utilised to identify the necessary parameters in Revit, both built-in and new. Most dimension parameters such as width, length, and area; constraint parameters such as level, base constraint, and top offset; and material parameters were already embedded in Revit and used for the identification process. Table 3 presents examples of the identification process for the wall family, and Figure 6 shows the properties of the wall family in Revit. A similar identification process was applied to all families in Revit.

Table 3. Identification of parameters in Revit.

Clause	Type	Marked-Up Phrase	Parameters of Wall Family
142(1)	<a>	The external wall is carried across the end of a separating wall	Wall Function
142(3)	<e>	An external wall of a building that is within the limits of the size indicated by the letter "x" in Part 1 of the Ninth Schedule to these by-laws	Classification.OmniClass.23.Number
158(1)	<r>	Carried down to a solid foundation	Base Offset
141(3)	<r>	Distance of not less than 225 millimetres measured at right angles to a such upper surface	Unconnected height
150(2)	<r>	Be completed enclosed	Room bounding
142(2)	<r>	Comply with any relevant requirements relating to the permitted limits of unprotected areas specified in the Sixth Schedule to these by-laws	Length
158(1)	<r>	Brickwork	Material
218	<e>	Load-bearing wall	Structural usage
142(3)	<r>	Be constructed so as to attain any FRP required by this Part	Fire resistance period (h)
222(3)	<e>	The building is so situated that such a side might consist entirely of any unprotected area	Wall distance to the relevant boundary

During the rule interpretation stage, the rules were analysed according to building type and space. For example, the compliance rules for dwellings differ from those for factories. The requirements for an occupied space and a circulation space also contain different sets of compliance rules. Therefore, the building types and building spaces were assigned specific names. Two classification systems were developed through Autodesk Classification Manager for the Revit® add-in. This tool provides a classification system database file for users to create parameters with a set of predetermined values in Revit. Figure 7 illustrates the uploaded database that could be applied to assign specific values to the parameters.

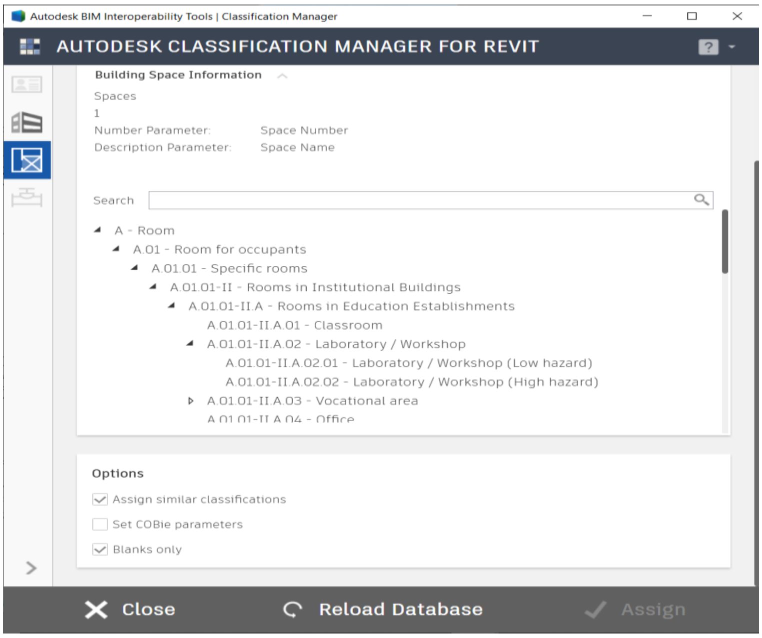


Basic Wall
Generic - 150mm

Walls (1)

Image	
Comments	
Mark	
Wall Function	Compartment Wall
Wall Function (Number)	D
Phasing	
Data	
Fire Resistance Period (hr)	2.000000
Classification of exterior finish	Class 0
Insulation period (mins)	60.000000
Performance index BSD 476 test...	
Standard specifications for insul...	Clause 11c section 3 BS 476

Figure 6. Wall family properties pane in Revit.



Building Space Information

Spaces
1
Number Parameter: Space Number
Description Parameter: Space Name

- A - Room
  - A.01 - Room for occupants
    - A.01.01 - Specific rooms
      - A.01.01-II - Rooms in Institutional Buildings
        - A.01.01-II.A - Rooms in Education Establishments
          - A.01.01-II.A.01 - Classroom
          - A.01.01-II.A.02 - Laboratory / Workshop
            - A.01.01-II.A.02.01 - Laboratory / Workshop (Low hazard)
            - A.01.01-II.A.02.02 - Laboratory / Workshop (High hazard)
          - A.01.01-II.A.03 - Vocational area
          - A.01.01-II.A.04 - Office

Options

☑ Assign similar classifications
☐ Set COBie parameters
☑ Blanks only

Figure 7. Uploaded database for room family in Revit.

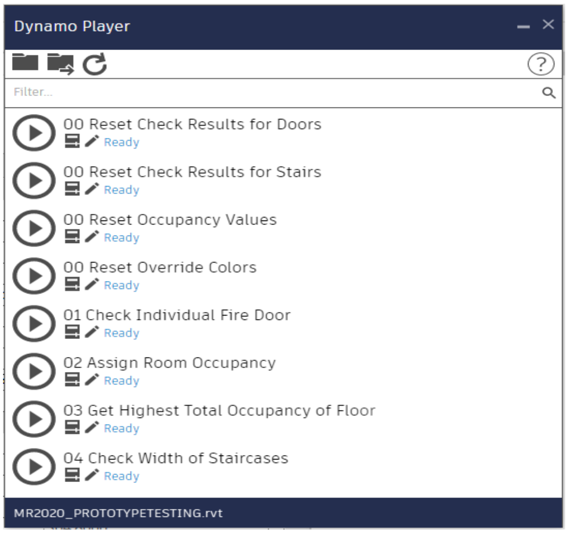
4.3. Encoding Process/Proof-of-Concept Prototype

Before encoding the rules into Revit, pseudocodes were developed to represent the algorithm for BIMSMACC. Two types of pseudocode modules were created: Search Modules and Checking Modules. Eleven Search Modules were created to process arrays in the schedules provided in the fire safety regulations, while nineteen Checking Modules were created to demonstrate the rule execution and reporting procedure. The Search Modules could be used to search for and select the appropriate requirements according to the building purpose groups, building spaces, and building elements. The Checking Modules were divided according to the building elements. In developing the modules, the topics, comparators, and target values from the marked-up phrases were utilised to influence the compliance results as either 'PASS' or 'FAIL'.

All modules contain sub-modules to divide the task systematically. A Checking Module known as PC.00 was regarded as the mainline module for the compliance checking procedure, with subordinate modules such as PC.01, PC.02, and PC.03. These subordinate modules also contain further subordinate modules, such as PC.06.01 and PC.06.02 as subordinate modules of PC.06. The Search Modules are utilised in the Checking Modules to select appropriate requirements from the schedules in fire safety regulations. The BIM properties acted as the global data for all modules, and variables were drawn from these data to influence the compliance checking results. Desk checking was then conducted for all pseudocode modules to ensure the accuracy of the algorithms' logic. Two programmers validated the pseudocodes through desk checking, and these experts confirmed that the pseudocodes were logically designed.

In encoding the rules in Revit, a prototype was developed through Dynamo to demonstrate a portion of the rule execution and reporting process for BIMSMACC. The pseudocodes acted as guidelines in developing the Dynamo scripts. The scope of this prototype was limited to fire doors and staircases in the BIM model. The compliance checking process in the prototype was driven by the manipulation of families and parameters in the BIM model. Seven families were manipulated for the prototype: project information, floor plan, room, wall, stairs, runs, and door families. Besides utilising the previously identified parameters, new parameters were purposely created to display the compliance results.

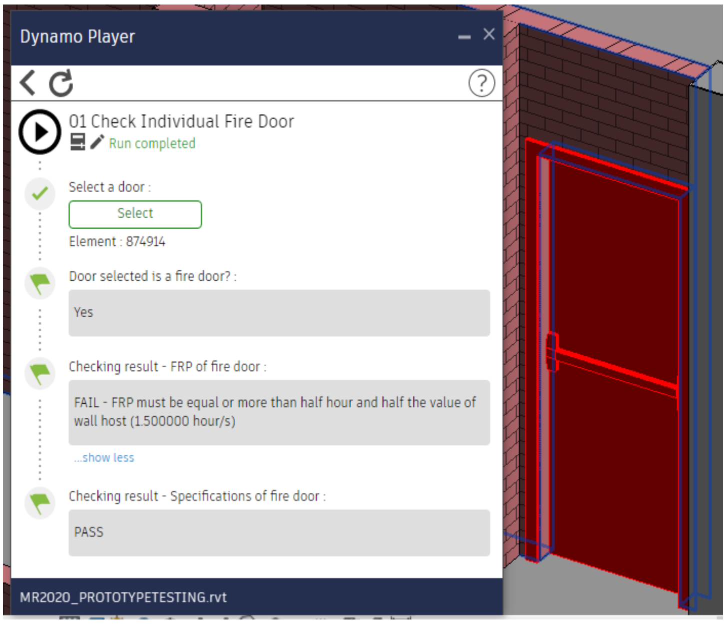
A total of eight Dynamo scripts were created where users can run the scripts through the Dynamo Player in Revit, as shown in Figure 8. The reset scripts were created for validation purposes, where the values of relevant parameters could be reset to iterate the prototype testing. A user could run the '01 Check Individual Fire Door' script to demonstrate the automated compliance checking process for fire doors. The script would check the fire resistance period (FRP) and standard specifications of a door family. The script would first prompt the user to select the door and then assess the type of door through its OmniClass Number. If the user selects a fire door, the script will extract the door's fire resistance period and compare it with the required fire resistance period of the building, wall host, or building spaces. For compliance checking of the fire door's specifications, the script will compare the value of the Standard Specifications parameter of the door family with the required standard. The compliance results will then be reflected in the Dynamo Player, the properties pane of the door, and the 3D view. Figures 9 and 10 provide examples of a non-compliant fire door.



00 Reset Check Results for Doors
Ready
00 Reset Check Results for Stairs
Ready
00 Reset Occupancy Values
Ready
00 Reset Override Colors
Ready
01 Check Individual Fire Door
Ready
02 Assign Room Occupancy
Ready
03 Get Highest Total Occupancy of Floor
Ready
04 Check Width of Staircases
Ready

MR2020_PROTOTYPETESTING.rvt

Figure 8. Scripts in Dynamo Player for BIMSMACC.



Dynamo Player

01 Check Individual Fire Door

Run completed

Select a door :

Select

Element : 874914

Door selected is a fire door? :

Yes

Checking result - FRP of fire door :

FAIL - FRP must be equal or more than half hour and half the value of wall host (1.500000 hour/s)

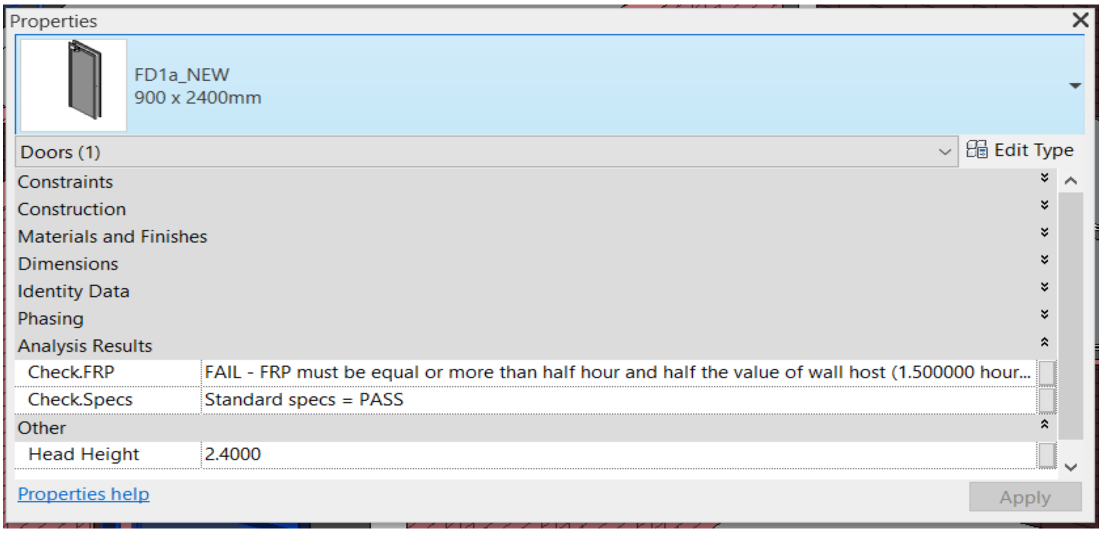
...show less

Checking result - Specifications of fire door :

PASS

MR2020_PROTOTYPETESTING.rvt

Figure 9. Automated compliance checking process for a fire door.



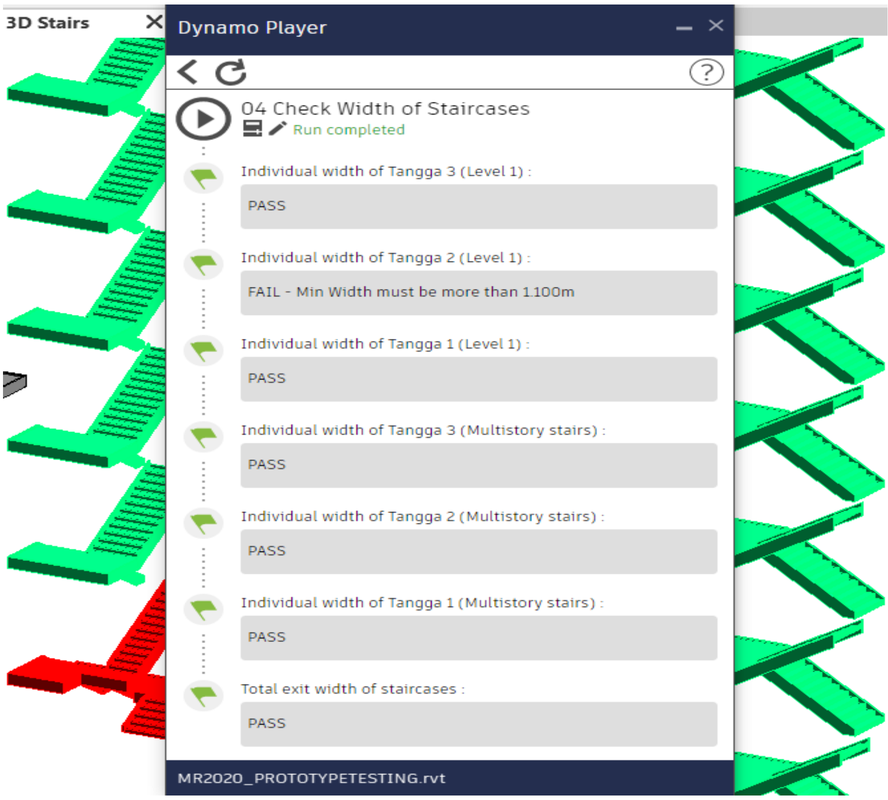
Check.FRP FAIL - FRP must be equal or more than half hour and half the value of wall host (1.500000 hour...

Check.Specs Standard specs = PASS

Head Height 2.4000

Figure 10. Compliance results in the properties pane of the door.

Three scripts were developed for the compliance checking process of staircases to avoid a large number of nodes in a single script. First, a user can run the '02 Assign Room Occupancy' script to calculate the number of occupants for each building space. Next, the user can run the '03 Get Highest Total Occupancy of Floor' script to obtain the highest total occupancy of that floor level. Subsequently, the user can run the '04 Check Width of Staircase' script. The width of individual staircases will be checked by extracting the actual run width of the staircase and comparing it with the requirements. The script would also check the total exit width of staircases. The script would acquire a set of data representing a schedule in fire safety regulations to determine the exit capacity of staircases. By utilising the highest total occupancy of floor level and the exit capacity, the script would then calculate the required exit width. The value would be applied to check the total exit width of staircases. The demonstration of compliance results for staircases is shown in Figure 11.



04 Check Width of Staircases
Run completed

Individual width of Tangga 3 (Level 1) :
PASS

Individual width of Tangga 2 (Level 1) :
FAIL - Min Width must be more than 1.100m

Individual width of Tangga 1 (Level 1) :
PASS

Individual width of Tangga 3 (Multistory stairs) :
PASS

Individual width of Tangga 2 (Multistory stairs) :
PASS

Individual width of Tangga 1 (Multistory stairs) :
PASS

Total exit width of staircases :
PASS

MR2020_PROTOTYPETESTING.rvt

Figure 11. Automated compliance checking process for staircases.

4.4. Validation of Framework and Prototype

The validation of the framework was conducted to evaluate the experts' level of agreement on the importance, completeness, and reliability of the framework. The validation results indicate high mean ratings ranging from 4.09 to 4.37 out of 5.00, as shown in Table 4. All experts believed that the framework has the potential to be implemented as an automated system as a tool to assist designers. One of the experts suggested that the implementation of the automated system could be more interactive, for instance, in a Virtual Reality environment; this would assist not only designers but also clients to visualise the checking process. Furthermore, another expert believed that this could also be used by the fire authority to check the minimum compliance before approving the building design.

Table 4. Mean ratings for the validation of framework and prototype.

Criteria	Description	Mean
	Framework	
Importance	To assess the importance and relevance of the elements and approaches proposed in the framework	4.35
Completeness	To assess the accuracy of the elements presented in the framework	4.09
Reliability	To assess the extendibility of the framework in developing the automated system	4.37
	Prototype	
Usability	To assess the overall display of functions, required inputs, generated outputs, and the ability to use the prototype	4.77
Accuracy	To assess the accuracy of the checking process and compliance results	4.96

Table 4. *Cont.*

Criteria	Description	Mean
Reliability	To assess the reliability of the prototype to be applied to other scopes and extended into a comprehensive automated system	4.88
Improvement to existing practice	To assess the efficiency of the prototype to assist designers during the BIM modelling process.	4.79

All the experts also believed that all operators of the RASE methodology—that is, the <r>, <a>, <s>, and <e> operators—are equally important in interpreting the meaning of each clause. One expert considered the RASE methodology as a means of transforming the conventional form of the clauses into a digital library version; this would give designers a better understanding of the meaning of each clause. All experts stressed that although compliance could be checked automatically, there should be an allowance to alter the final compliance results based on fire safety experts' judgement, especially for informative clauses.

In the validation process, the prototype's features and functions were first demonstrated to the experts. Next, the experts were invited to test the prototype through the Dynamo Player in Revit. Subsequently, they provided feedback and filled out the questionnaires. The prototype was validated based on its usability, accuracy, reliability, and improvement to existing practice. The mean ratings for the validation of the prototype were very high, between 4.79 to 4.96 out of 5.00 (Table 4). High mean ratings for all criteria affirmed that the automated system proposed in this research has a shallow learning curve, is easy to use, is accessible, and is an improvement over the existing manual process. All agreed that the prototype is user-friendly as it operates in Revit without requiring any new software application. The suggestions and feedback were considered in outlining the contributions and limitations of this research, as well as the prospects for future research.

5. Discussions

Although this study adopted the methods introduced by other past studies, this paper emphasised on establishing and structuring the conditions of the methods as well as addressing any limitations that occurred. For the categorisation process, in the original work [83], the filter system commenced with the first filter categorisation followed by the second filter categorisation. This study introduced a supplementary procedure before initiating the first filter categorisation where clauses containing references to other clauses or regulations should be identified, ensuring that the interrelationship between those rules remain intact. The dependence on other requirements was also acknowledged in another study [85].

This study also established the conditions for declarative, informative, and remaining clauses. These conditions are presented in Table 5 along with examples of excerpts from the clauses in the Selangor Uniform Building (Amendment) (No. 2) By-Laws 2012. The declarative clauses should be categorised by identifying clauses containing geometrical or definite rules such as the dimension of spacing for seatings and the availability of a fire protection system. For informative clauses, two conditions were established. First, this category should represent any clauses that contain ambiguous terminologies or arrangements. Keywords such as "where appropriate", "suitable", adequate", "as far as practical", "furthermost point of floor", and "as may be required by fire authority" are indicators of ambiguity in the clauses. Second, the informative clauses should represent clauses that provide options for designers. For example, the keywords "either", "or", and "where practical" refer to the consideration of designers in deciding the best design solution. The remaining clauses should then be able to meet one of the three criteria: explaining the terminology used in other clauses, providing general provisions for other clauses, or providing specific calculation methods for other clauses.

Table 5. Examples of clauses.

Type	Condition	Examples of By-Laws
Declarative clauses	Contain geometrical rules	*By-law 184(1)(a)—The spacing of rows of seats from back-to-back shall be not less than 825 millimetres, nor less than 675 millimetres plus the sum of the thickness of the back and inclination of the back.*
	Contain definite rules	*By-law 153(1)—All lift lobbies shall be provided with smoke detectors.*
		By-law 163—Fire doors including frames shall be constructed per MS 1073.
Informative clauses	Contain ambiguous terminologies or arrangements	*By-law 171(1)—Where appropriate, horizontal exits may be provided in lieu of other exits.*
		By-law 236—Places constituting special hazards or risk due to the nature of storage, trade, occupancy, or size shall be required to be protected by fixed installations, protective devices, systems, and special extinguishers as may be required by the D.G.F.R.
	Provide options for designers	*By-law 196(5)—Where natural ventilation is impractical, smoke lobbies and fire-fighting access lobbies may be ventilated utilizing a vertical shaft or mechanically pressurised.*
Remaining clauses	Contain definitions of terminologies used in other by-laws	*By-law 133—In this Part and Part VIII, unless the context otherwise requires, "exit door" means a door from a storey, flat, or room that gives access from such storey, flat, or room onto an exit route;*
	Contain general provisions or guidelines for other by-laws	*By-law 145(b)—Any reference to the Sixth Schedule to these by-laws shall be construed as referring to the provisions of Part I of the Schedule together with, at the option of the persons intending to erect the building, either the provisions of Part II, Part III, or Part IV of the Schedule.*
	Contain method of calculation or measurement	*By-law 165(1)—The travel distance to an exit shall be measured on the floor or other walking surface along the centre line of the natural path of travel, starting 0.300 metres from the most remote point of occupancy, curving around any corners or obstructions with 0.300-metre clearance therefrom and ending at the storey exit. Where measurement includes stairs, it shall be taken in the plane of the trend noising.*

For the decomposition of semantics in the clauses, previous work revealed three limitations of the RASE methodology: first, the relationship between individual rules was absent; second, the <a> and <s> operators created an unnecessary repetition; third, the marking-up process for <s> and <e> operators was redundant [43]. These limitations were taken into consideration and the conditions were restructured in this study. The first limitation was addressed in the categorisation process, where any clause that contains reference to other requirements should be marked. For the second limitation, in the original work, the <s> operator was applied to mark any phrases containing definitions or alternative terms for <a> marked-up phrases [16]. Another study argued that the identification of the <s> operator created redundancy as this operator had a similar function to the <a> operator [43]. Although the redundancy was undeniable, this was not considered a limitation in this study. The identification of the <s> operator was necessary to maintain the originality of the clauses. Differentiating the <a> and <s> operators was challenging as most clauses contain sub-rules; thus, iteration occurred in refining their functions.

Therefore, this study established that the <a> operator should be applied to mark the main object in a clause while other objects should be marked with the <s> operator. In the original work, the <e> operator was the opposite of the <a> operator. The definition of the <e> operator was further defined in this study, where it should also be an exception to the <r> and <s> operators. Although previous work identified the <e> operator as redundant, the identification of <e> marked-up phrases should be conducted to retain the interrelationships between the rules.

After the marking-up process, the clauses should be formalised in a tabular manner to identify the topic, comparator, and target value. The original work suggested selecting commonly used terms from a restricted AEC dictionary to determine the topics of the phrases [16]. Although determining the topics was argued to be too repetitive [43], it was considered relevant in this study. Since the terms for the topic might vary according to country, the topics should be determined based on clauses or regulations that contain definitions of common terminologies. For example, in the case of the Selangor Uniform Building (Amendment) (No. 2) By-Laws 2012, by-law 133 was used as a point of reference to determine the topics of the phrases. To standardise the identification of comparators, <r> marked-up phrases should apply "equal", "not equal", "more than", and "less than" comparators while <a>, <s>, and <e> operators should apply "include" or "exclude" comparators. As implemented in the case studies, the topics, comparators, and target values should be organised to restructure the fire safety clauses. Any exceptions to the rules should also be included in the restructured clauses, improving the restructuring process introduced in the previous work [41]. This inclusion is vital to preserve the relationship between the main rules and exceptions. In any case where the restructured clauses disclose any discrepancies with the original meaning of clauses, the whole decomposition process should be iterated until accuracy can be achieved.

In the case study, interview sessions were conducted to interpret the informative clauses. From the interview sessions, this study concluded that the informative clauses should be open-ended as these clauses highly depend on the judgement of designers and the relevant authorities. These judgements vary between construction projects, designers' experience, and jurisdictions in different countries. Thus, an automated system that incorporates informative clauses should allow designers and relevant authorities to override the compliance results.

For the identification of BIM properties, the OmniClass™ classification system should be regarded as a reliable classification system in standardizing the naming of components or building elements. As implemented in the case studies, custom classification systems were built to define the building types and spaces that could be adopted in other regulations. This study provided a proof-of-concept prototype in Revit through Dynamo scripts. Through this method, designers could directly use the compliance results from the Dynamo scripts to rectify their BIM models. Alternatively, the result could be exported into a report by creating Schedules in Revit for doors and staircases. These Schedules could then be exported into Microsoft Excel. The flexibility for users to run the scripts either in Dynamo Player or the built-in Dynamo tool proved that this proposed automated system could accommodate high user accessibility to manage the rules. The development of scripts according to the pseudocode modules demonstrates the separation of rule management and rule execution. Although the proposed system was tested in Revit, the pseudocodes and scripts could be replicated in other BIM software applications.

6. Conclusions and Recommendations

This paper describes a user-friendly approach to developing an automated system for compliance checking in a BIM-based environment. The automated system was implemented on a total of 256 fire safety clauses applicable in Malaysia and validated by Malaysian fire safety experts.

The intellectual contribution of the study is threefold. First, the semi-automated translation process for building regulations addresses the issue of limited involvement from AEC experts in past studies. The overall rule translation process consists of five approaches: categorisation, decomposition through semantic mark-up RASE methodology, interviews with relevant AEC experts, development of pseudocodes, and the encoding process through Dynamo. The combination of these techniques presents the novelty of this study and eliminates the limitations found in existing studies as it encourages greater participation of AEC professionals, compared with software developers. Second, the identification of necessary BIM properties in native file format overcomes the challenges in preparing the model for IFC data. Through Revit file format representation, designers could simultaneously employ the automated system during the design process without having the burden of converting the BIM models into IFC representation. This approach makes it practicable for designers in using the system. Third, the study provides the necessary information to develop BIMSMACC consisting of the BIM properties, a set of pseudocodes, and a prototype in Revit. The BIM properties provide guidelines to designers in equipping their BIM models for compliance checking. The pseudocodes could be considered a digital representation of the fire safety clauses in the Selangor Uniform Building (Amendment) (No 2) By-Laws 2012; they act as a basis for automation of the compliance checking process in a BIM-based environment.

This paper focuses on the implementation of the proposed framework for the compliance checking of BIM models against the fire safety regulations in Malaysia. It was found that while the clauses could be semantically decomposed and interpreted, inputs from designers and the fire authority should not be dismissed in determining the final compliance results. The results are still subjected to the judgement of designers during the design process and to approval from the fire authority. Through the proof-of-concept prototype, this study highlights the main function of the proposed system as assisting the designers and not substituting the responsibilities of the fire engineers. Based on the validation by the experts, the proposed framework and prototype are accurate and reliable.

The limitation of this study includes the research scope, whereby the selected clauses were derived from the Selangor Uniform Building (Amendment) (No. 2) By-Laws 2012. Other regulations such as Malaysian Standards, British Standards, and internal policies by the fire authority were not included during the interpretation process. For the BIM model, the BIM properties defined for this study were limited to the application in Revit only. Another limitation is that due to the subjectivity in informative clauses, the compliance results should be subjected to the judgement of designers during the design process and the local authority's approval. Thus, it is to be highlighted that the main purpose of this proposed system is to provide a tool in assisting designers and not to substitute the responsibilities of the fire engineers.

Future works include the extension of the prototype to become a comprehensive application that can accommodate all fire safety clauses in Malaysia. Other regulations such as Malaysian Standards and discretionary requirements from the fire authority can also be added to the system. The BIM properties defined in this study could be converted to the IFC data model to accommodate other BIM platforms. The framework could also be extended to other types of building regulations and not limited to the fire safety regulations only.

Author Contributions: Conceptualization, A.S.I., K.N.A., N.A.I. and M.A.K.; methodology, A.S.I., K.N.A., N.A.I., N.T.A.-A. and M.A.K.; validation, A.S.I., K.N.A. and N.A.I.; formal analysis, A.S.I., K.N.A., N.A.I. and M.A.K.; writing—original draft preparation, A.S.I. and M.A.K.; writing—review and editing K.N.A. and N.T.A.-A., visualization, A.S.I., K.N.A., N.A.I., N.T.A.-A. and M.A.K.; supervision, M.A.K., N.A.I. and M.A.K.; project administration, M.A.K. and K.N.A. All authors have read and agreed to the published version of the manuscript.

Funding: Universiti Teknologi Malaysia (UTM) Research Grant VOT numbers: 5F584 and 05E79.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data sets used during and/or analysed during the current study are available from the corresponding author upon reasonable request.

Acknowledgments: We would like to thank all experts for their contributions to the research.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

2D	Two-dimensional
AEC	Architecture, Engineering, and Construction
BIM	Building Information Modelling
BIMSMACC	BIM-based Automated System for Malaysian Code Compliance Checking
BREAAM	Building Research Establishment Environmental Assessment Method
CAD	Computer-Aided Design
CSH	Code For Sustainable Homes
DSR	Design Science Research
EDM	Express Data Manager
FOL	First-Order Logic
FRP	Fire Resistance Period
IE	Information Extraction
IFC	Industry Foundation Classes
IFC	Industry Foundation Classes
ITr	Information Transformation
KBVL	Kbim Visual Language
LSC	Life Safety Codes
NLP	Natural Language Processing
OTTV	Overall Thermal Transfer Value
RASE	Requirement, Applicability, Selection, And Exception
RKM	Regulatory Knowledge Model
SMC	Solibri Model Checker
SWRL	Semantic Web Rule Language
VCCL	Visual Code Checking Language
VPL	Visual Programming Language

References

1. Eastman, C.; Teicholz, P.; Sacks, R.; Liston, K. *BIM Handbook A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers, and Contractors*, 3rd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2018.
2. Nawari, N.O. *Building Information Modeling Automated Code Checking and Compliance Processes*, 1st ed.; CRC Press: Boca Raton, FL, USA, 2018.
3. Eastman, C.; Lee, J.-M.; Jeong, Y.-S.; Lee, J.-K. Automatic rule-based checking of building designs. *Autom. Constr.* **2009**, *18*, 1011–1033. [CrossRef]
4. Delis, E.A.; Delis, A. Automatic Fire-Code Checking Using Expert-System Technology. *J. Comput. Civ. Eng.* **1995**, *9*, 141–156. [CrossRef]
5. Smith, R.L. EXPOSURE: An expert system fire code. *Fire Technol.* **1991**, *27*, 145–159. [CrossRef]
6. Heikkila, E.J.; Blewett, E.J. Using Expert Systems to Check Compliance with Municipal Building Codes. *J. Am. Plan. Assoc.* **1992**, *58*, 72–80. [CrossRef]
7. Myers, L.; Snyder, J.; Chirica, L. Database Usage in a Knowledgebase Environment for Building Design. *Build. Environ.* **1992**, *27*, 231–241. [CrossRef]
8. Khemlani, L. CORENET e-PlanCheck: Singapore's Automated Code Checking System. 2005. Available online: http://www.aecbytes.com/buildingthefuture/CORENETePlanCheck.htm (accessed on 25 March 2023).
9. Dimyadi, J.; Amor, R. Automated building code compliance checking—Where is it at? *Proc. CIB WBC* **2013**, *6*, 1–14.
10. Ding, L.; Drogemuller, R.; Rosenman, M.; Marchant, D.; Gero, J. Automating Code Checking for Building Designs-DesignCheck 2006, 1–16. Available online: https://ro.uow.edu.au/engpapers/4842 (accessed on 8 April 2023).
11. Lee, J. *Automated Checking of Building Requirements on Circulation Over a Range of Design Phases, College of Architecture*; Georgia Institute of Technology: Atlanta, Georgia, 2010.
12. Lê, M.A.T.; Mohus, F.; Kvarsvik, O.K.; Lie, M. *The HITOS Project—A Full Scale IFC Test, eWork and eBusiness in Architecture, Engineering and Construction*; CRC Press: Boca Raton, FL, USA, 2006; pp. 191–195.

13. Ghannad, P.; Lee, Y.-C.; Dimyadi, J.; Solihin, W. Automated BIM data validation integrating open-standard schema with visual programming language. *Adv. Eng. Inform.* **2019**, *40*, 14–28. [CrossRef]
14. Dimyadi, J.; Pauwels, P.; Amor, R. Modelling and accessing regulatory knowledge for computer-assisted compliance audit. *J. Inf. Technol. Constr.* **2016**, *21*, 317–336.
15. Preidel, C.; Borrmann, A. *BIM-Based Code Compliance Checking, Building Information Modeling*; Springer: New York, NY, USA, 2018; pp. 367–381.
16. Hjelseth, E.; Nisbet, N. Exploring Semantic Based Model Checking. In Proceedings of the CIB W78 2010: 27th International Conference, Cairo, Egypt, 16–19 November 2010; pp. 1–11.
17. Zhang, J.; El-Gohary, N.M. Semantic NLP-Based Information Extraction from Construction Regulatory Documents for Automated Compliance Checking. *J. Comput. Civ. Eng.* **2016**, *30*, 04015014. [CrossRef]
18. Preidel, C.; Borrmann, A. Towards Code Compliance Checking on the Basis of a Visual Programming Language. *ITcon* **2016**, *21*, 402–421.
19. Zhang, S.; Boukamp, F.; Teizer, J. Ontology-based semantic modeling of construction safety knowledge: Towards automated safety planning for job hazard analysis (JHA). *Autom. Constr.* **2015**, *52*, 29–41. [CrossRef]
20. Zhang, J.; El-Gohary, N.M. A Prototype System for Fully Automated Code Checking, eWork and eBusiness in Architecture. In Proceedings of the 11th European Conference on Product and Process Modelling (ECPPM 2016), Limassol, Cyprus, 7–9 September 2016; pp. 535–542.
21. Preidel, C.; Borrmann, A. Refinement of the Visual Code Checking Language for an Automated Checking of Building Information Models Regarding Applicable Regulations. In Proceedings of the ASCE International Workshop on Computing in Civil Engineering 2017, Seattle, WA, USA, 25–27 June 2017; pp. 157–165.
22. Kim, H.; Lee, J.K.; Shin, J.; Choi, J. BIM-Supported Visual Language to Define Building Design Regulations. In *Proceedings of the 22nd International Conference on Computer-Aided Architectural Design Research in Asia: Protocols, Flows and Glitches, Hong Kong, China, 9–15 April 2017*; Janssen, P., Loh, P., Raonic, A., Schnabel, M.A., Eds.; The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA): Hong Kong, China, 2017; pp. 603–617.
23. Katsigarakis, K.I.; Giannakis, G.I.; Lilis, G.N.; Rovas, D.V. An IFC data preparation workflow for building energy performance simulation. In Proceedings of the 2019 European Conference on Computing in Construction, Chania, Crete, Greece, 10–12 July 2019; pp. 164–171. [CrossRef]
24. Pauwels, P.; Zhang, S.; Lee, Y.-C. Semantic web technologies in AEC industry: A literature overview. *Autom. Constr.* **2017**, *73*, 145–165. [CrossRef]
25. Wu, J.; Zhang, J. New Automated BIM Object Classification Method to Support BIM Interoperability. *J. Comput. Civ. Eng.* **2019**, *33*, 04019033. [CrossRef]
26. Koo, B.; Shin, B. Applying novelty detection to identify model element to IFC class misclassifications on architectural and infrastructure Building Information Models. *J. Comput. Des. Eng.* **2018**, *5*, 391–400. [CrossRef]
27. Monteiro, A.; Martin, J. A survey on modeling guidelines for quantity takeoff-oriented BIM-based design. *Autom. Constr.* **2013**, *35*, 238–253. [CrossRef]
28. Preidel, C.; Borrmann, A. Automated Code Compliance Checking Based on a Visual Language and Building Information Modeling. In Proceedings of the 32nd International Symposium of Automation and Robotics in Construction, Oulu, Finland, 15–18 June 2015; pp. 1–8.
29. BCA. CORENET—About Us. 2021. Available online: https://www.corenet.gov.sg/general/about-us.aspx (accessed on 2 March 2023).
30. GSA. *BIM Guide 02 Spatial Program Validation*; U.S.G.S. Administration, Ed.; U.S. General Services Administration: Washington, DC, USA, 2020.
31. Invicara, BIM Assure Deliver Complete, Reliable BIM Data. 2020. Available online: https://invicara.com/technology/bim-assure/ (accessed on 13 December 2022).
32. Daima. Daima Streamlines the Plan-Checking Process. 2020. Available online: https://daima.squarespace.com/ (accessed on 18 January 2023).
33. Shih, S.-Y.; Sher, W.; Giggins, H. Assessment of the building code of Australia to inform the development of BIM-enabled code checking system. In Proceedings of the 19th World Building Congress: Construction and Society, Brisbane, QLD, Australia, 5–9 May 2013; pp. 1–12.
34. Lee, Y.-C.; Eastman, C.; Lee, J.-K. Validations for ensuring the interoperability of data exchange of a building information model. *Autom. Constr.* **2015**, *58*, 176–195. [CrossRef]
35. Greenwood, D.; Lockley, S.; Malsane, S.; Matthews, J. Automated compliance checking using building information models. In Proceedings of the Construction, Building and Real Estate Research Conference, Paris, France, 2–3 September 2010; pp. 363–371.
36. Kincelova, K.; Boton, C.; Blanchet, P.; Dagenias, C. BIM-Based Code Compliance Checking for Fire Safety in Timber Buildings: A Comparison of Existing Tools. *Can. Soc. Civ. Eng. Can.* **2019**, CON226-01–CON226-10. Available online: http://amz.xcdsystem.com/A464D029-DBD8-FB0C-B034B891CCEED78C_abstract_File10551/PaperPDFversion_226_0603014558.pdf (accessed on 8 April 2023).
37. Daima. Frequently Asked Questions. 2020. Available online: https://daima.squarespace.com/faq (accessed on 22 February 2023).

38. Fenves, S.J.; Gaylord, E.H.; Goel, S.K. *Decision Table Formulation of the 1969 AISC Specification*; University of Illinois: Urbana, IL, USA, 1969.
39. Tan, X.; Hammad, A.; Fazio, P. Automated Code Compliance Checking for Building Envelope Design. *J. Comput. Civ. Eng.* **2010**, *24*, 203–211. [CrossRef]
40. Wix, J.; Nisbet, N.; Liebich, T. Using Constraints to Validate and Check Building Information Models. In *eWork and eBusiness in Architecture, Engineering and Construction*; Zarli, A., Scherer, R., Eds.; CRC Press: Boca Raton, FL, USA, 2009; pp. 467–476.
41. Hjelseth, E.; Nisbet, N. Capturing Normative Constraints by Use of the Semantic Mark-Up Rase Methodology. In Proceedings of the CIB W78-W102 2011: International Conference, Sophia Antipolis, France, 26–28 October 2011; pp. 1–10.
42. Beach, T.H.; Kasim, T.; Li, H.; Nisbet, N.; Rezgui, Y. Towards automated compliance checking in the construction industry. In Proceedings of the Database and Expert Systems Applications: 24th International Conference, DEXA 2013, Prague, Czech Republic, 26–29 August 2013; pp. 366–380.
43. Macit, S.; İlal, M.E.; Günaydın, H.M. Computer representation of building codes for automated compliance checking. *Autom. Constr.* **2017**, *82*, 43–58. [CrossRef]
44. Getuli, V.; Ventura, S.M.; Capone, P.; Ciribini, A.L.C. BIM-based Code Checking for Construction Health and Safety. *Procedia Eng.* **2017**, *196*, 454–461. [CrossRef]
45. Dimyadi, J.; Clifton, C.; Spearpoint, M.; Amor, R. Computerizing Regulatory Knowledge for Building Engineering Design. *J. Comput. Civ. Eng.* **2016**, *30*, C4016001. [CrossRef]
46. Dimyadi, J.; Solihin, W.; Eastman, C.; Amor, R. Integrating the BIM Rule Language into Compliant Design Audit Processes. In Proceedings of the 33rd CIB W78 Conference: IT in Construction, Brisbane, Australia, 31 October–2 November 2016; pp. 1–10.
47. Solihin, W.; Eastman, C. A Knowledge Representation Approach in BIM Rule Requirement Analysis using the Conceptual Graph. *ITcon* **2016**, *21*, 370–402.
48. Fan, S.-L.; Chi, H.-L.; Pan, P.-Q. Rule checking Interface development between building information model and end user. *Autom. Constr.* **2019**, *105*, 102842. [CrossRef]
49. Lee, H.; Lee, J.-K.; Park, S.; Kim, I. Translating building legislation into a computer-executable format for evaluating building permit requirements. *Autom. Constr.* **2016**, *71*, 49–61. [CrossRef]
50. Kim, H.; Lee, J.-K.; Shin, J.; Choi, J. Visual language approach to representing KBimCode-based Korea building code sentences for automated rule checking. *J. Comput. Des. Eng.* **2019**, *6*, 143–148. [CrossRef]
51. Zhou, P.; El-Gohary, N.M. Text and Information Analytics for Fully Automated Energy Code Checking. In *Project Management and BIM for Sustainable Modern Cities*; GeoMEast: 2018. Sustainable Civil Infrastructures; Shehata, M., Rodrigues, F., Eds.; Springer: Cham, Switzerland, 2019; pp. 196–208.
52. Garrett, J.H.; Palmer, M.E.; Demir, S. Delivering the Infrastructure for Digital Building Regulations. *J. Comput. Civ. Eng.* **2014**, *28*, 167–169. [CrossRef]
53. Zhang, J.; El-Gohary, N.M. Automated Information Extraction from Construction-Related Regulatory Documents for Automated Compliance Checking. In Proceedings of the CIB W78-W102 2011 International Conference, Sophia Antipolis, France, 26–28 October 2011; pp. 1–10.
54. Zhou, P.; El-Gohary, N.M. Semantic-based Text Classification of Environmental Regulatory Documents for Supporting Automated Environmental Compliance Checking in Construction. In Proceedings of the Construction Research Congress 2014, Atlanta, Georgia, 19–21 May 2014; pp. 897–906.
55. Zhou, P.; El-Gohary, N.M. Ontology-based automated information extraction from building energy conservation codes. *Autom. Constr.* **2017**, *74*, 103–117. [CrossRef]
56. Zhang, J.; El-Gohary, N.M. Integrating semantic NLP and logic reasoning into a unified system for fully-automated code checking. *Autom. Constr.* **2017**, *73*, 45–57. [CrossRef]
57. Song, J.; Kim, J.; Lee, J.-K. NLP and Deep Learning-based Analysis of Building Regulations to support Automated Rule Checking System. In Proceedings of the 35th International Symposium on Automation and Robotics in Construction (ISARC, 2018), Berlin, Germany, 20–25 July 2018; p. 133.
58. Song, J.; Lee, J.-K.; Choi, J.; Kim, I. Deep learning-based extraction of predicate-argument structure (PAS) in building design rule sentences. *J. Comput. Des. Eng.* **2020**, *7*, 563–576. [CrossRef]
59. Xue, X.; Zhang, J. Building Codes Part-of-Speech Tagging Performance Improvement by Error-Driven Transformational Rules. *J. Comput. Civ. Eng.* **2020**, *34*, 04020035. [CrossRef]
60. Zhang, K. *Visual Languages and Applications*; Springer: New York, NY, USA, 2007.
61. Preidel, C.; Daum, S.; Borrmann, A. Data retrieval from building information models based on visual programming. *Vis. Eng.* **2017**, *5*, 18. [CrossRef]
62. Kim, H.; Lee, J.K.; Shin, J.; Kim, J. Visual Language-based approach for the Definition of Building Permit related rules. In Proceedings of the 34th International Symposium on Automation and Robotics in Construction (ISARC 2017), Taipei, Taiwan, 28 June–1 July 2017.
63. Kim, J.; Lee, J.-K.; Shin, J.; Kim, H. Classification of objects and their properties defined in Korea Building Act to translate into KBimCode and their application. In Proceedings of the 34th International Symposium on Automation and Robotics in Construction (ISARC 2017), Taipei, Taiwan, 28 June–1 July 2017; pp. 1–8.

64. Zhou, H.; Lee, S.; Ying, H. VPL-Based Code Translation for Automated Compliance Checking of Building Envelope Energy Efficiency. In Proceedings of the Construction Research Congress 2018, New Orleans, LA, USA, 2–4 April 2018; pp. 1–12.
65. Ghannad, P.; Lee, Y.-C.; Dimyadi, J.; Solihin, W. Modularized BIM Data Validation Framework Integrating Visual Programming Language with LegalRuleML. In *Advances in Informatics and Computing in Civil and Construction Engineering*; Mutis, I., Hartmann, T., Eds.; Springer: Cham, Switzerland, 2019; pp. 85–93.
66. Choi, J.; Kim, I. Development of an Open BIM-based Legality System for Building Administration Permission Services. *J. Asian Archit. Build. Eng.* **2018**, *14*, 577–584. [CrossRef]
67. Hitzler, P.; Krötzsch, M.; Rudolph, S. *Foundations of Semantic Web Technologies*; CRC Press: Boca Raton, FL, USA, 2010.
68. Yurchyshyna, A.A.; Zucker, C.F.; Thanh, N.L.; Lima, C.; Zarli, A. Towards an Ontology-Based Approach for Conformance Checking Modeling in Construction. In Proceedings of the 24th W78 Conference on Information Technology for Construction, Maribor, Slovenia, 27–29 June 2007; pp. 195–202.
69. Yurchyshyna, A.; Faron-Zucker, C.; Thanh, N.L.; Zarli, A. Towards an Ontology-based Approach for Formalisation of Expert Knowledge in Conformity Checking Model in Construction. In Proceedings of the 7th European Conference on Product and Process Modelling (ECPPM), Lisbon, France, 23–28 June 2008; pp. 447–456. Available online: https://www.researchgate.net/publication/300040540_Towards_an_ontology-based_approach_for_formalizing_expert_knowledge_in_the_conformity-checking_model_in_construction (accessed on 8 April 2023).
70. Zhong, B.; Gan, C.; Luo, H.; Xing, X. Ontology-based framework for building environmental monitoring and compliance checking under BIM environment. *Build. Environ.* **2018**, *141*, 127–142. [CrossRef]
71. Xu, X.; Cai, H. Semantic approach to compliance checking of underground utilities. *Autom. Constr.* **2020**, *109*, 103006. [CrossRef]
72. Zhong, B.T.; Ding, L.Y.; Luo, H.B.; Zhou, Y.; Hu, Y.Z.; Hu, H.M. Ontology-based semantic modeling of regulation constraint for automated construction quality compliance checking. *Autom. Constr.* **2012**, *28*, 58–70. [CrossRef]
73. Zhang, S.; Boukamp, F.; Teizer, J. Ontology-Based Semantic Modeling of Safety Management Knowledge. In Proceedings of the 2014 International Conference on Computing in Civil and Building Engineering, Orlando, FL, USA, 23–25 June 2014; pp. 2254–2262.
74. Shih, S.-Y.; Sher, W. Development of Building Information Modelling Enabled Code Checking Systems for Australia. In Proceedings of the 17th International Symposium on Advancement of Construction Management and Real Estate; Wang, J., Ding, Z., Zou, L., Zuo, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 1003–1010. Available online: https://econpapers.repec.org/bookchap/sprsprbok/978-3-642-35548-6.htm (accessed on 8 April 2023).
75. Kasim, T. *BIM-Based Smart Compliance Checking to Enhance Environmental Sustainability*; Cardiff School of Engineering; Cardiff University: Cardiff, UK, 2015.
76. Zhang, R.; El-Gohary, N.M. A Clustering Approach for Analyzing the Computability of Building Code Requirements. In Proceedings of the Construction Research Congress 2018, New Orleans, LA, USA, 2–4 April 2018; pp. 86–95.
77. Statsbygg, SIMBA 1.3. 2020. Available online: https://sites.google.com/view/simba-bim-krav/simba-1-3 (accessed on 1 March 2023).
78. Plume, J.; Mitchell, J. Collaborative design using a shared IFC building model—Learning from experience. *Autom. Constr.* **2007**, *16*, 28–36. [CrossRef]
79. BCA, Building Information Modeling (BIM) e-Submission. 2021. Available online: https://www.corenet.gov.sg/general/building-information-modeling-(bim)-e-submission.aspx (accessed on 12 March 2023).
80. Hevner, A.; Chatterjee, S. *Design Research in Information Systems Theory and Practice*; Springer: New York, NY, USA, 2010.
81. Peffers, K.; Tuunanen, T.; Rothenberger, M.A.; Chatterjee, S. A Design Science Research Methodology for Information Systems Research. *J. Manag. Inf. Syst.* **2007**, *24*, 45–77. [CrossRef]
82. Weber, S. Comparing Key Characteristics Of Design Science Research As An Approach And Paradigm. In Proceedings of the Pacific Asia Conference on Information Systems (PACIS 2012), Ho Chi Minh City, Vietnam, 11–15 July 2012; p. 180.
83. Malsane, S.; Matthews, J.; Lockley, S.; Love, P.E.D.; Greenwood, D. Development of an object model for automated compliance checking. *Autom. Constr.* **2015**, *49*, 51–58. [CrossRef]
84. Solihin, W.; Eastman, C. Classification of rules for automated BIM rule checking development. *Autom. Constr.* **2015**, *53*, 69–82. [CrossRef]
85. Nawari, N.O. A Generalized Adaptive Framework for Automating Design Review Process: Technical Principles. In *Advances in Informatics and Computing in Civil and Construction Engineering*; Springer: Chem, Switzerland, 2019; pp. 405–414.
86. Autodesk. Features. 2021. Available online: https://www.autodesk.com/products/revit/features?plc=RVT&term=1-YEAR&support=ADVANCED&quantity=1#internal-link-top-features (accessed on 13 June 2022).
87. Weir, T.S.; Richardson, J.D.; Wing, E.; Harrington, D.J. *Harrington, Mastering Revit Structure*, 1st ed.; Wiley Publishing: Indianapolis, IN, USA, 2009.
88. PWD, Objektif Penubuhan Unit Projek BIM. 2013. Available online: https://www.jkr.gov.my/prokom/index.php?option=com_content&view=article&id=312&Itemid=478&lang=ms (accessed on 25 January 2023).
89. Latiffi, A.A.; Mohd, S.; Kasim, N.; Fathi, M.S. Building Information Modeling (BIM) Application in Malaysian Construction Industry. *Int. J. Constr. Eng. Manag.* **2013**, *2*, 1–6.

90. MyBIM. BIM Module Training. 2020. Available online: https://mybim.cidb.gov.my/product-category/training/bim-module-training/ (accessed on 5 February 2023).
91. Hjelseth, E. *Foundations for BIM-Based Model Checking Systems*; Department of Mathematical Sciences and Technology, Norwegian University of Life Sciences: Oslo, Norway, 2015.