Journal of Construction Engineering, Management & Innovation
2022 5(1):01-14
DOI 10.31462/jcemi.2022.01001014

RESEARCH ARTICLE

Development of BIM software with quantity take-off and visualization capabilities

Faruk Ergen, Önder Halis Bettemir

Inonu University, Department of Civil Engineering, Malatya, Türkiye

Article History

Received 02 November 2021
Accepted 15 March 2022

Keywords

Building Information Modeling
Python programming language
SQLite
3D visualization
Ursina engine

Abstract

Building Information Modeling (BIM) prepares quantity take-off of construction items, helps the management of the design and construction process and prepares 3D visualization of the construction phases. BIM offers more control over the construction tasks and improves the efficiency. Construction professionals are aware of the benefits of BIM however, the utilization of BIM is still not common yet because of the BIM software cost and training requirement of the staff. In this study, open-source BIM software is developed by Python programming language. The software is capable of preparation of 2-dimensional drawings, 3D visualization as well as execution of quantity take-off computations of formwork and concrete. The codes are developed in Python language by using SQlite library and Ursina engine. A graphical user interface is formed by Python language for the execution of 2 Dimensional drawings and entries of the coordinates of the structural elements as well as attribute data. 3D coordinates of the structural elements are computed by using the joint coordinates of the structural elements. 3D visualizations of the structural elements are performed by the Ursina engine. The connections and intersections of the joints and structural elements are stored in a database that is formed by SQlite. The voids are successfully computed and formwork and concrete quantities are computed. The open-source BIM software would have low investment and operational costs for the construction firms and would increase BIM usage. The utilization of the software would decrease the manmade errors during the quantity take-off preparation step.

1. Introduction

Construction sector works on project-based businesses therefore; each construction endeavor has unique computations such as preparation of drawings, quantity take-off, execution of structural analysis, preparation of construction schedule, and the construction of the project. The mentioned tasks require the cooperation of multi-disciplinary professionals and the management of data flow. On the other hand, construction sector is very competitive in nature so that the construction firms have to execute the tasks quickly and economically. These goals create significant stress on the project stakeholders which makes the execution of the tasks vulnerable to manmade errors.

BIM is a convenient tool for the construction firms to manage the data flow between the project participants. In addition to this, one BIM software can handle many tasks which can be done by several softwares. Preparation of technical

Correspondence Önder Halis Bettemir onder.bettemir@inonu.edu.tr

drawings, quantity take-off, and the schedule of the construction tasks can be given as examples that are performed by different software. Execution of construction management tasks on one software has several benefits for the contractors and the engineers as it reduces the risk of malfunctioning data extraction, incompatibilities caused by the version differences, and rework of the prepared documents.

The mentioned benefits of BIM increased its popularity which escalates the academic and practical studies on BIM. Kalfa [1] mentioned the application areas of BIM for academic, industry, and certification purposes. Omaran et al. [2] integrated life cycle cost analysis with BIM by a game engine. The 4D simulation of the construction process represents both the construction and operation costs of the constructed utility. A survey is prepared among the users of the application. The survey results show that the participants declared that the developed application mainly affects the decision they made. Delgado et al. [3] examined the utilization of Augmented Reality (AR) and Virtual Reality (VR) by the Architecture, Engineering and Construction (AEC) sectors. Benefits, complexities, difficulties, and deficiencies of the state-of-the-art AR and VR technology are discussed by making four group discussions. The study concludes that the implementations of AR and VR technologies are inadequate for the AEC sector. Ozcan-Deniz [4] examined the possible applications of VR by interviewing 18 construction firms. Sarı and Pekeriçli [5] investigated and compared the legal issues on the implementation of BIM in Turkey, USA, and UK. Temel and Başağa [6] analyzed the code compliance checking from the data extracted from Industry Foundation Classes (IFC) file format. They have reached the fact that BIM models and software are suitable for the detection of code violations without human endeavor.

BIM and VR integrated systems are used for occupational health and safety education and the prevention of occupational accidents. Dhalmahapatra et al. [7] prepared an educational and experimental tool to reduce the number of occupational accidents due to the utilization of mobile cranes. Albahbah et al. [8] investigated the literature on the VR and concluded that visualization has potential benefits for the construction sector in terms of occupational health and safety, conflict detection, and construction management education.

Adobe Acrobat Reader enables the addition of a 3D model for version 9 or later. Nadeem et al. embedded 3D models which are prepared on Autodesk 3DSMax to Adobe Acrobat Reader pdf files. The mentioned process allows execution of rotation as well as zooming in and out on the 3D drawings of a building by Adobe Acrobat Reader software [9]. Sacks and Barak [10] concluded that 3D modeling decreases the design and drawing errors, provides shorter delivery time, increases efficiency and reduces costs.

Millman and Aivazis illustrate that Python programming language become a more suitable environment for the programming of engineering problems. The ability of handling complex numbers, the easiness of plotting and graphics make the programming language more popular [11]. Quraishi and Dhapekar investigated the difficulty level of programming in Python and stated that instead of using spreadsheets for the necessary computations for construction site management, the same computations can be programmed in Python language easily by civil engineers [12]. Sarvade and Pore [13] developed software by Python which makes reinforced concrete beam design. The study briefly introduces the NumPy, SciPy, Pandas, and Matplotlib libraries of Pythons programming language.

Quantity take-off of construction items is error-prone and requires exhausting manual computations. Bettemir [14] proposed a methodology to reduce the computational demand and workload of quantity take-off computations of excavation, concreting, and formwork. Khosakitchalert et al. [15-18] used Dynamo version 1.3.3.4111 to prepare quantity take-off of construction items such as walls and floor coverings.

Lee et al. [19] estimated the QTO and construction cost of tiling construction items by automated inference process. The inference mechanism is fed by expert opinion based on the work conditions. The ontological inference tool considers room usage, building element, finishing thickness, tile size, type, thickness, tiling method, tiling material type, joint width and joint material type. Consequently, the workmanship estimation thus the construction cost estimation turns out to be better than the standard cost estimation.

Choi et al. [20] prepared a schematic QTO estimation process for the preliminary design stage of the buildings. The data is extracted from IFC model to Open BIM software. The process executes the formation of schematic estimation, verification of data, verification of data quality and computation of QTO.

Liu et al. [21] developed an ontology-based semantic approach for QTO. The system requires the formation of entity relationships among construction items and related materials and workmanship which constitutes the ontology. Host relationships such as a wall can host a wall are defined, as well as geometric and topological relationships. Voids of quantity take-off are computed by database queries. The proposed system enabled faster and more accurate QTO.

Olsen and Taylor [22] examined the benefits of the utilization of BIM software by making a survey. The respondents stated the advantages of BIM as fast QTO preparation, and more accurate QTO than manually prepared QTO. In addition to this, BIM assists risk identification if it is implemented early enough. The disadvantages of implementation of BIM are stated as the complexity of BIM software and the possibility of entering outdated data.

Kim et al. [23] investigated the discrepancies in QTO amounts caused by the definition of object models. Compositely modeled object and individually modeled object definitions are investigated and the study concludes that compositely modeled object definitions may cause up to 6% deviation from the exact QTO value. Individually modeled object definition contains more detail and thus provides better results.

Liu et al. [24] proposed a new knowledge model-based quantity take-off model to eliminate the inability of the state-of-the-art BIM software which does not enclose all the necessary data for precise automatic quantity take-off (QTO). The study suggests defining different types of slab-wall; slab-beam; slab-column and slab-wall joint connections. Consequently, better QTO estimations are obtained.

The quantity take-off computations executed on BIM are finished in shorter durations when compared with manual computations. On the other hand, manual computations are exact if they are executed without mistakes. Some of the construction methods may affect the quantity take-off computations. Excavation by ripper blade and hydraulic hammer can be given as an example for this situation [25]. Preparing the quantity take-off by BIM can help compare different construction methods. The utilization of BIM has some obstacles. The differences between the codes, documentation, and language make development of a universal BIM significantly difficult.

There are free open source applications that contribute at certain stages in the BIM process. To illustrate Free BIM Viewer and BIM Vision are two examples for free open-source BIM solutions. In addition to this, BIMx is an application developed by Graphisoft to display 3D models, while FreeCAD is an open-source application coded in the python programming language. However, the mentioned software cannot compete with Revit as a stand-alone BIM solution. The mentioned solutions require important improvement in terms of quantity take-off and code compliance. Therefore the BIM software development endeavor mentioned in this study aims to present a free open source stand-alone BIM solution in the future.

BIM software cannot prepare exact quantity take-off because the robust detection of intersections of structural and architectural elements is not achieved by state-of-the-art BIM software. Therefore, assumptions are made when computing the voids due to the intersection of the structural elements. Intersections of column-beam, column-slab and beam-slab can be given as examples for this situation. The exact intersection area cannot be computed due to the complex intersection geometry and simplifications are made which causes generalization errors. Quantity take-off values prepared by BIM software are not exact because of the aforementioned simplifications. In this study, BIM software is developed in Python programming language, which has 3D visualization as well as quantity take-off abilities. The aforementioned simplifications on the computation of quantity take-off are avoided and the intersected sections are determined exactly. As a result of this, the voids are computed accurately.

The development of the BIM software is explained in the methodology section, a case study is made to compare the quantity take-off results of the developed software and the existing software, finally, the results are discussed and concluded.

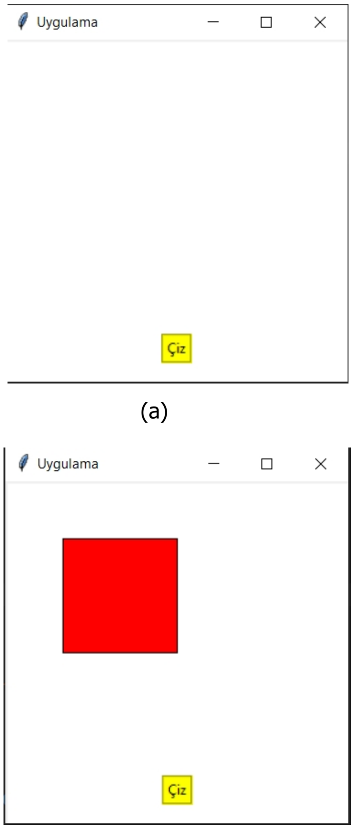
2. Methodology

In the methodology section computation steps of the quantity take-off procedure and the programming of the quantity take-off computations in Python programming language are explained.

2.1. Development of BIM software

The quantity take-off procedure is executed on developed BIM software which allows data entry by mouse and keyboard, the definition of the structural elements, and computation of quantity take-off of formwork and concrete. Data entry for the definition of structural elements is performed by a Graphical User Interface (GUI) which is programmed by Tkinter library. This library is compatible with Python programming language and uses the Frame() class to organize tool groups. Structural elements are drawn by Canvas() library which can draw shapes on a window. Buttons, text boxes, and dropdown lists and their controls are programmed by the control commands of the Python programming language.

Fig. 1a illustrates a GUI which is prepared by Tkinter. The window has Çiz (draw) button which draws a red rectangle when pressed. Fig. 1b shows the drawn red rectangle.

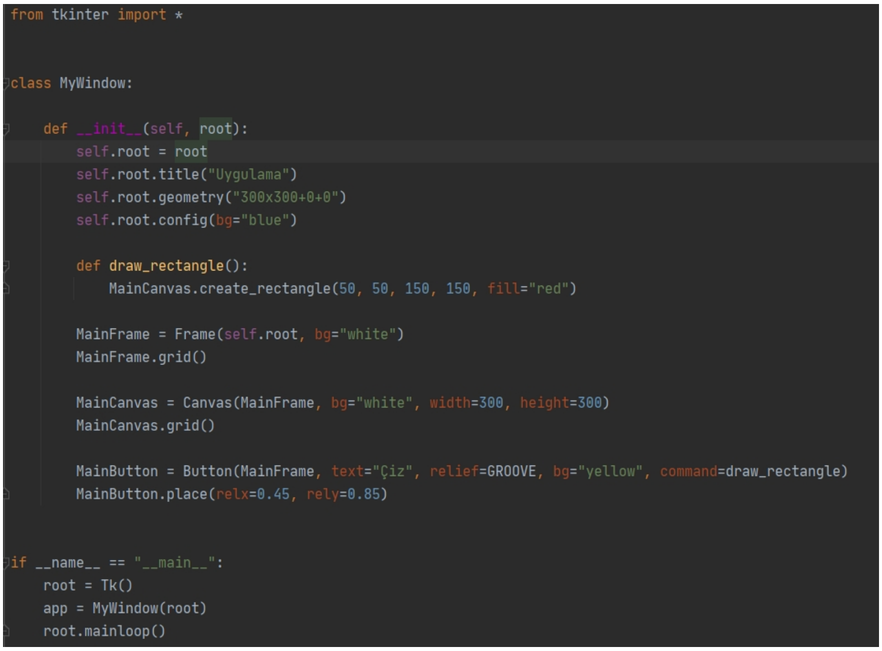


(a)

(b)

Fig.1. (a) GUI prepared by Tkinter is shown. (b) represents the drawn red rectangle when the button is pressed

Fig. 2 represents the programming code written in Python for the control commands on the button to draw the rectangle. In addition to this *MainCanvas* and *MainButton* tools are tied to *MainFrame* to organize the structure of the window. Function *draw_rectangle*() is called by the control of the MainButton and *root.mainloop()* function provides an infinite loop that allows continuous drawing. The user defines the structure by clicking the locations of the joints with the mouse. The coordinates of the clicked pixel are retrieved by the *event.x* and *event.y* commands. The obtained pixel coordinates are used as input for the *create_rectangle()* function. The predefined methods *scan_mark()* and *scan_dragto()* of *Canvas()* class are used for the pan command of the drawing window.



```
from tkinter import *


class MyWindow:

    def __init__(self, root):
        self.root = root
        self.root.title("Uygulama")
        self.root.geometry("300x300+0+0")
        self.root.config(bg="blue")

        def draw_rectangle():
            MainCanvas.create_rectangle(50, 50, 150, 150, fill="red")

        MainFrame = Frame(self.root, bg="white")
        MainFrame.grid()

        MainCanvas = Canvas(MainFrame, bg="white", width=300, height=300)
        MainCanvas.grid()

        MainButton = Button(MainFrame, text="Çiz", relief=GROOVE, bg="yellow", command=draw_rectangle)
        MainButton.place(relx=0.45, rely=0.85)


if __name__ == "__main__":
    root = Tk()
    app = MyWindow(root)
    root.mainloop()
```

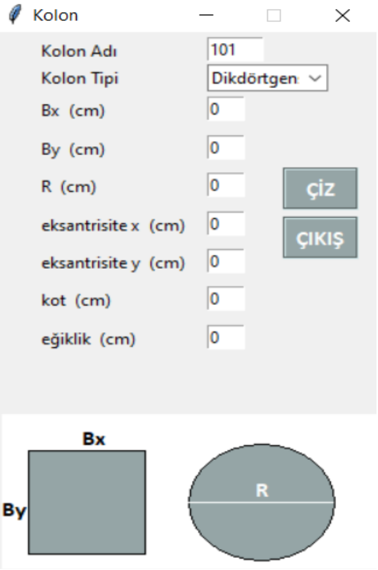
Fig. 2. Sample codes written for the button control to draw a rectangle

Columns and beams should be placed on the axis lines. The user defines the number of the axis and the spacing between the axis lines. The software draws the axis lines in x and y directions. After the definition of the axis lines, the user can draw the structural elements.

Columns, beams, and slabs are defined by drawing. The drawing of each type of load-bearing element is executed by pressing a particular button. The user initially declares the type of the structural element. The software waits until the user clicks the mouse on the drawing palette to define the position of the structural element. When the user clicks on the drawing canvas, a GUI window is shown to the user to define the dimensions and characteristics of the structural element. Figure 3 shows a GUI for the definition of the shape, dimensions, eccentricity, and vertical position of the column. Convenient GUI forms are prepared for the data entry of required features.

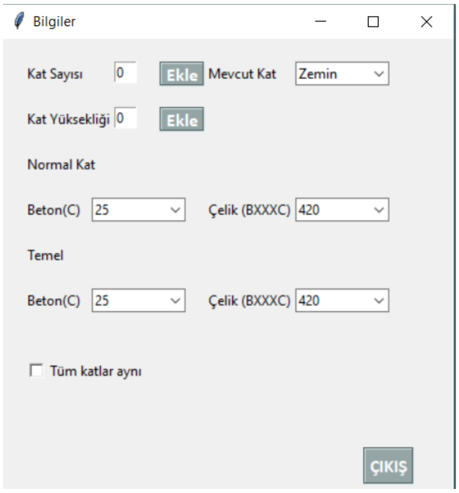
Required data might differ according to the specifications of the structural element. For instance, circular columns require only the definition of the radius; on the other hand, rectangular columns require width and height information. Preparation of separate GUIs for such instances is possible but it leads to the preparation and management of significantly many GUI forms and control buttons. Therefore, a specific GUI is not designed unless it is mandatory. The GUI controls the data integrity by assuring the necessary information is entered. The GUI shown in Figure 3 compels the user to enter data for the Bx and By textboxes. If a circular column was selected, then the GUI would request a nonzero integer value for the radius textbox.

Some of the attribute data of the structural elements are retrieved by the same data types. The overlapping data for each structural element are combined and these data are obtained by a separate GUI. The attribute data of drawn structural elements are entered by a developed GUI furnished by textboxes, buttons and dropdown lists. Concrete type and class, reinforcement type and floor height can be defined for the selected structural element. A key ID number is given for each Structural element and they are given a code according to their structural behavior. Columns are numbered starting with “S”, beams are numbered starting with “K” and slabs are numbered starting with “D”. The prepared GUI for the entry of attribute data is shown in Fig. 4.



Kolon Adı: 101
Kolon Tipi: Dikdörtgen
Bx (cm): 0
By (cm): 0
R (cm): 0
eksantrisite x (cm): 0
eksantrisite y (cm): 0
kot (cm): 0
eğiklik (cm): 0

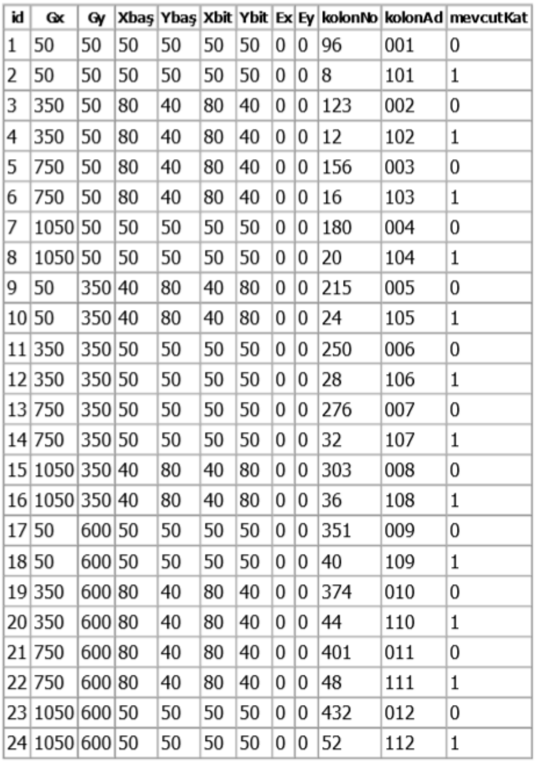
Fig. 3. The GUI to draw a column



Kat Sayısı: 0
Mevcut Kat: Zemin
Kat Yüksekliği: 0
Normal Kat
Beton(C): 25
Çelik (BXXXC): 420
Temel
Beton(C): 25
Çelik (BXXXC): 420
□ Tüm katlar aynı

Fig. 4. Prepared GUI for the entry of attribute data

BIM software should possess a database that is capable of executing the data entry, data storage, data query, data manipulation, data editing, and formation of the relationship between the data tables. The database of the developed BIM software is developed by a database engine designed by Sqlite. Attribute data and the dimensions of the structural elements are stored in the database of the BIM software. The database allows deleting and editing of the defined structural elements. Fig. 5 illustrates the database table of slabs.



id	Gx	Gy	Xbaş	Ybaş	Xbit	Ybit	Ex	Ey	kolonNo	kolonAd	mevcutKat
1	50	50	50	50	50	50	0	0	96	001	0
2	50	50	50	50	50	50	0	0	8	101	1
3	350	50	80	40	80	40	0	0	123	002	0
4	350	50	80	40	80	40	0	0	12	102	1
5	750	50	80	40	80	40	0	0	156	003	0
6	750	50	80	40	80	40	0	0	16	103	1
7	1050	50	50	50	50	50	0	0	180	004	0
8	1050	50	50	50	50	50	0	0	20	104	1
9	50	350	40	80	40	80	0	0	215	005	0
10	50	350	40	80	40	80	0	0	24	105	1
11	350	350	50	50	50	50	0	0	250	006	0
12	350	350	50	50	50	50	0	0	28	106	1
13	750	350	50	50	50	50	0	0	276	007	0
14	750	350	50	50	50	50	0	0	32	107	1
15	1050	350	40	80	40	80	0	0	303	008	0
16	1050	350	40	80	40	80	0	0	36	108	1
17	50	600	50	50	50	50	0	0	351	009	0
18	50	600	50	50	50	50	0	0	40	109	1
19	350	600	80	40	80	40	0	0	374	010	0
20	350	600	80	40	80	40	0	0	44	110	1
21	750	600	80	40	80	40	0	0	401	011	0
22	750	600	80	40	80	40	0	0	48	111	1
23	1050	600	50	50	50	50	0	0	432	012	0
24	1050	600	50	50	50	50	0	0	52	112	1

Fig. 5. Database table developed on Sqlite

Coordinates of the structural elements are transferred to the database once the user confirms the drawing of the structural element. A unique key id is assigned to the structural element and the data of the structural element is stored in the corresponding data column. The recently drawn element is stored at the bottom line of the data table. Required data can be retrieved from the database for the 3D visualization or quantity take-off computations.

Database is formed by *sqlite.connect()* function. The retrieval of data from the database table is executed by the same command. Changes and manipulation on the data table can be saved by the *commit()* function. The *cursor()* function allows retrieving data from a particular row of the data table. SQL queries are performed by the *execute()* function.

2.2. Quantity take-off computations

Quantity take-off computations are performed according to the unit price analysis. Quantity take-off calculation methods of the Republic of Türkiye Ministry of Environment and Urbanization are implemented. In this study, the quantity take-off computations of formwork and concrete are programmed on the developed BIM software. The amount of formwork is computed by considering the area of concrete that is to be covered by the formwork. The structural elements intersect with other structural elements which makes the area computations complex. Column-beam, column-slab and beam-slab intersection areas have to be determined and reduced from the gross formwork area.

Column-beam intersections are detected by examining the start and end nodes of the beams and columns. If node ids are matched, the examined beam and column intersect. The beam wholly intersects the column but the thickness of the slab is reduced from the depth of the beam. The reduced depth is multiplied by the width of the beam and the obtained value is voided from the gross formwork area.

Column-slab intersections are detected by comparing the node id of the columns and every node id of the slabs. If node ids are matched, the investigated column and slab intersect. Eccentricities of both the column and slabs are considered and the length of the intersection is computed. Slab thickness is multiplied by the intersection length and the obtained area is voided from the gross formwork area of the column.

If side of the column is larger than the beam, the base of the column might penetrate into the slab area. This situation is examined by comparing the position of the column with the border of the slab. The eccentricity of the column is used to determine whether the base of the column intersects with the slab. If the column intersects the slab; the intersection area is computed by detecting the intersection points of the slab with the rectangle surrounding the column. The formed area is voided from the gross formwork area of the slab.

Beam-slab intersections are detected by matching the pair of start node and end node of the beam with the node ids of the beams. If id couples are matched, the examined beam and slab intersect. Beam intersects the slab throughout its entire span. The thickness of the slab is reduced from the thickness of the beam and the obtained value is multiplied by the span of the beam. This gives the side area of the formwork for the beam. Two sides of the beam are computed by checking any slab intersection. The bottom portion of the formwork does not have any voids. The aforementioned operations are implemented for each beam-slab intersection and the total voids are computed. The obtained voids are subtracted from the gross formwork area of the beams.

3. Case study

Developed BIM software is tested on two floors reinforced concrete building. The building has four and three-axis in the x and y directions respectively. There is a half-circle balcony on the east side of the building. Fig. 6 represents the drawn columns, beams and slabs. In addition to this, ladders are also drawn separately.

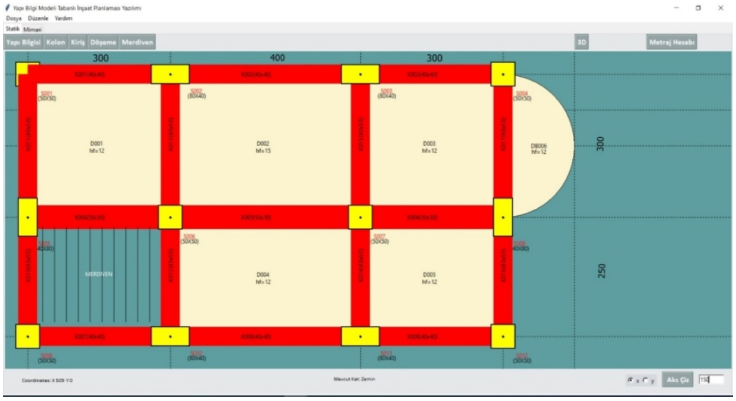


Fig. 6. Slab-plan drew by the developed BIM software.

The BIM software visualizes the structure as shown in Fig. 7. Computation of the 3D coordinates of the structural elements is executed as explained in the methodology chapter. Then the structural objects are visualized by the Tkinter library and Ursina engine which are compatible with Python programming language.

In the developed program, 2D drawings can be exported in dxf format. The floor plan drawing of second floor shown in Fig. 8 is exported as dxf file. The exported file is opened in AutoCAD as shown in Fig. 9. In future work, Industry Foundation Classes (IFC) support will be added. Code checking, cost estimation, energy analysis, facility management and etc. will be taken into account for interoperability by considering the current IFC standard. The database management system is capable of executing operations such as deleting, updating, adding and querying. However, the database management system cannot import or export data file from other BIM software. The developers aim to provide this feature in future work.

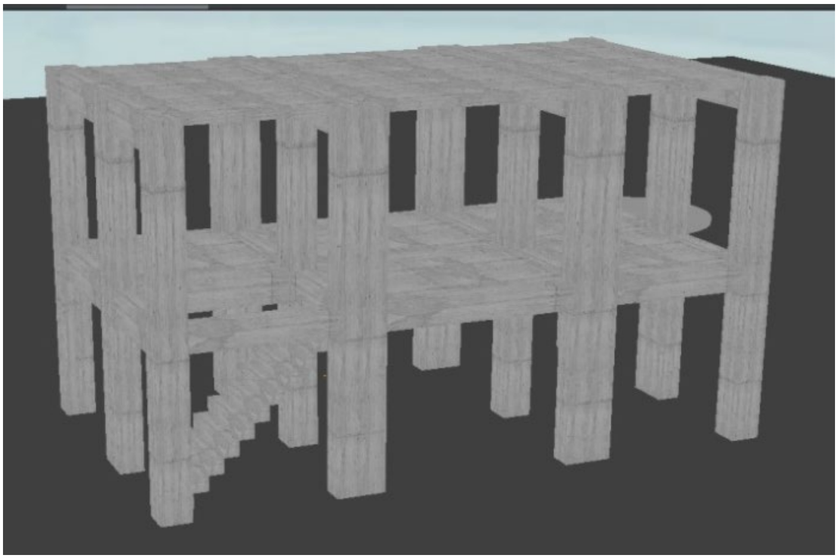


Fig. 7. 3-Dimensional representation of the structure.

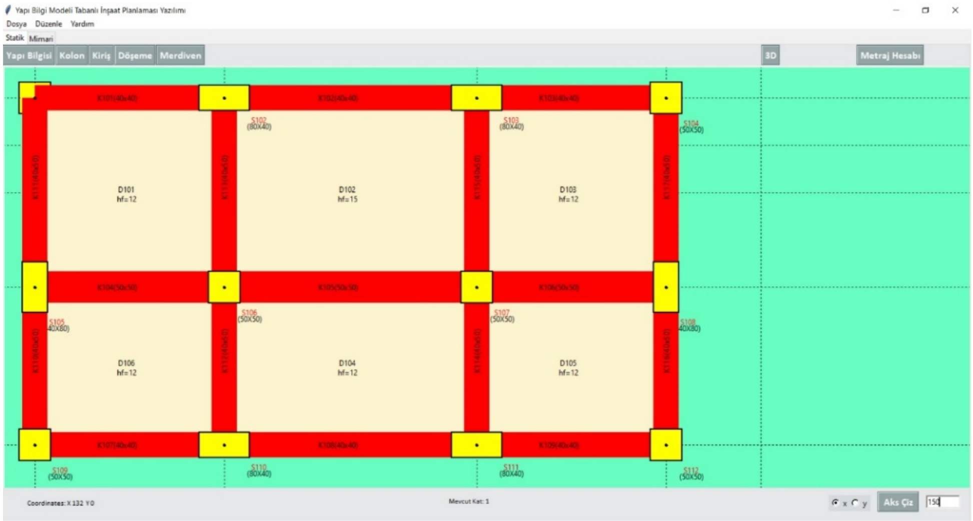
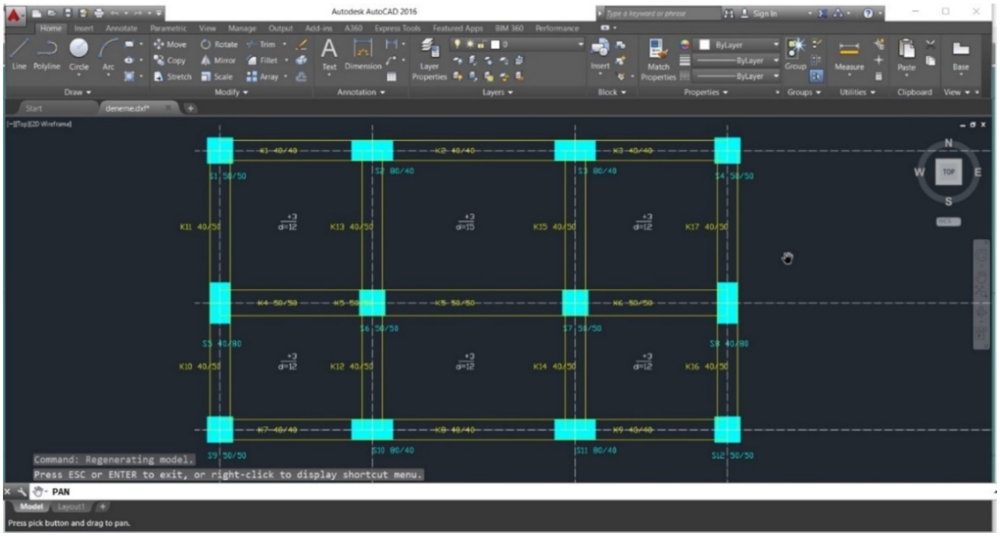


Fig. 8. Floor Plan drawing of the second floor prepared by the developed BIM software



```
Command: Regenerating model.
Press ESC or ENTER to exit, or right-click to display shortcut menu.
PAN
```

Press pick button and drag to pan.

Fig. 9. Exported dxf format of the drawing which is opened in AutoCAD

Manually computed quantity take-off of concrete and formwork for the entrance floor and first floors are given in Tables 1 and 2 respectively.

Table 1 and 2 illustrates the manually computed quantity take-off of the concrete and formwork amounts. The computations are performed according to the Republic of Türkiye Ministry of Environment and Urbanization. Voids are searched and examined attentively. Therefore the provided figures are exact.

The same quantity take-off computations are repeated by the developed BIM software. To compare the performance of the developed BIM software in terms of quantity take-off computations, quantity take-off computations of the same building are conducted by two local computer programs which can execute quantity take-off computations. The compared programs are represented as P1 and P2. The results of the quantity take-off computations are given in Table 3.

The numbers given in Table 3 are directly obtained from the quantity take-off report of the corresponding software. The percentage error of the quantity take-off values is given in Table 4.

4. Discussion of results

The quantity take-off computations are conducted by excluding the stairs and stairs landing. The obtained quantity take-off values of concrete and formwork include the remaining slabs, columns, beams, and the half-circular balcony. The quantity take-off computations are conducted according to the calculation methods of the Republic of Türkiye Ministry of Environment and Urbanization. In other words, voids that occurred because of column-beam, column-slab and beam-slab intersections are computed according to the directions of the Republic of Türkiye Ministry of Environment and Urbanization. The quantity take-off computation method may differ in different countries. Therefore, quantity take-off values of local software are compared.

Detection of voids and the computation of collision area or volume of intersected objects without human intervention is the most difficult part of the automated quantity take-off process. Quantity take-off software makes simplifications when forming the topology and detecting the neighborhood of the structural and architectural elements. Therefore, the intersection area of beam-column, beam-slab, and column-slab intersections cannot be computed exactly. This error propagates into void computations and prevents the preparation of robust quantity take-off computations.

Quantity take-off computations of the developed BIM software are compared with two local design software which also prepares the quantity take-off of the building.

Table 1. Quantity take-off values of concrete and formwork for the entrance floor

Ent. Floor		Dimensions (m)			Concrete Quantity Take-off (m^3)			Formwork Quantity Take-off (m^2)		
Slab	ID	Width	Length	Thick.	Gross Quantity	Void	Quantity	Gross Quantity	Void	Quantity
	DZ01	2.60	2.55	0.12	0.7956	0.0003	0.7953	6.63	0.0025	6.6275
	DZ02	3.60	2.55	0.15	1.377	0	1.377	9.18	0	9.18
	DZ03	2.60	2.55	0.12	0.7956	0.0003	0.7953	6.63	0.0025	6.6275
	DZ04	3.60	2.05	0.12	0.8856	0	0.8856	7.38	0	7.38
	DZ05	2.60	2.05	0.12	0.6396	0.0003	0.6393	5.33	0.0025	5.3275
Beam	ID	Width	Length	Thick	Quantity	Void	Quantity	Gross Quantity	Void	Quantity
	KZ01	2.35	0.4	0.4	0.376	0	0.376	2.82	0.282	2.538
	KZ02	3.2	0.4	0.4	0.512	0	0.512	3.84	0.48	3.36
	KZ03	2.35	0.4	0.4	0.376	0	0.376	2.82	0.282	2.538
	KZ04	2.55	0.5	0.5	0.6375	0	0.6375	3.825	0.306	3.519
	KZ05	3.5	0.5	0.5	0.875	0	0.875	5.25	0.945	4.305
	KZ06	2.55	0.5	0.5	0.6375	0	0.6375	3.825	0.612	3.213
	KZ07	2.35	0.4	0.4	0.376	0	0.376	2.82	0	2.82
	KZ08	3.2	0.4	0.4	0.512	0	0.512	3.84	0.384	3.456
	KZ09	2.35	0.4	0.4	0.376	0	0.376	2.82	0.282	2.538
	KZ10	1.85	0.4	0.5	0.37	0	0.37	2.59	0	2.59
	KZ11	2.35	0.4	0.5	0.47	0	0.47	3.29	0.282	3.008
	KZ12	2.05	0.4	0.5	0.41	0	0.41	2.87	0.246	2.624
	KZ13	2.55	0.4	0.5	0.51	0	0.51	3.57	0.6885	2.8815
	KZ14	2.05	0.4	0.5	0.41	0	0.41	2.87	0.492	2.378
	KZ15	2.55	0.4	0.5	0.51	0	0.51	3.57	0.6885	2.8815
	KZ16	1.85	0.4	0.5	0.37	0	0.37	2.59	0.222	2.368
	K117	2.35	0.4	0.5	0.47	0	0.47	3.29	0.282	3.008
Col	ID	Width	Length	Height	Quantity	Void	Quantity	Gross Quantity	Void	Quantity
	SZ01	0.5	0.5	3	0.75	0	0.75	6	0.372	5.628
	SZ02	0.8	0.4	3	0.96	0	0.96	7.2	0.574	6.626
	SZ03	0.8	0.4	3	0.96	0	0.96	7.2	0.574	6.626
	SZ04	0.5	0.5	3	0.75	0	0.75	6	0.372	5.628
	SZ05	0.4	0.8	3	0.96	0	0.96	7.2	0.668	6.532
	SZ06	0.5	0.5	3	0.75	0	0.75	6	0.9195	5.0805
	SZ07	0.5	0.5	3	0.75	0	0.75	6	0.9255	5.0745
	SZ08	0.4	0.8	3	0.96	0	0.96	7.2	0.686	6.514
	SZ09	0.5	0.5	3	0.75	0	0.75	6	0.36	5.64
	SZ10	0.8	0.4	3	0.96	0	0.96	7.2	0.544	6.656
	SZ11	0.8	0.4	3	0.96	0	0.96	7.2	0.568	6.632
	SZ12	0.5	0.5	3	0.75	0	0.75	6	0.372	5.628
							Σ22.9505			Σ157.4335

Table 2. Quantity take-off values of concrete and formwork for the first floor.

1st Floor		Dimensions (m)			Concrete Quantity Take-off (m^3)			Formwork Quantity Take-off (m^2)		
Slab	ID	Width	Length	Thick.	Gross Quantity	Void	Quantity	Gross Quantity	Void	Quantity
	D101	2.6	2.55	0.12	0.7956	0.0003	0.7953	6.63	0.0025	6.6275
	D102	3.6	2.55	0.15	1.377	0	1.377	9.18	0	9.18
	D103	2.6	2.55	0.12	0.7956	0.0003	0.7953	6.63	0.0025	6.6275
	D104	3.6	2.05	0.12	0.8856	0	0.8856	7.38	0	7.38
	D105	2.6	2.05	0.12	0.6396	0.0003	0.6393	5.33	0.0025	5.3275
	D106	2.6	2.05	0.12	0.6396	0.0003	0.6393	5.33	0.0025	5.3275
	Balcony	0	1.5	0.12	0.424103	0.0831	0.341003	4.099658	0	4.099658
Beam	ID	Width	Length	Thick.	Gross Quantity	Void	Quantity	Gross Quantity	Void	Quantity
	K101	2.35	0.4	0.4	0.376	0	0.376	2.82	0.282	2.538
	K102	3.2	0.4	0.4	0.512	0	0.512	3.84	0.48	3.36
	K103	2.35	0.4	0.4	0.376	0	0.376	2.82	0.282	2.538
	K104	2.55	0.5	0.5	0.6375	0	0.6375	3.825	0.612	3.213
	K105	3.5	0.5	0.5	0.875	0	0.875	5.25	0.945	4.305
	K106	2.55	0.5	0.5	0.6375	0	0.6375	3.825	0.612	3.213
	K107	2.35	0.4	0.4	0.376	0	0.376	2.82	0.282	2.538
	K108	3.2	0.4	0.4	0.512	0	0.512	3.84	0.384	3.456
	K109	2.35	0.4	0.4	0.376	0	0.376	2.82	0.282	2.538
	K110	1.85	0.4	0.5	0.37	0	0.37	2.59	0.222	2.368
	K111	2.35	0.4	0.5	0.47	0	0.47	3.29	0.282	3.008
	K112	2.05	0.4	0.5	0.41	0	0.41	2.87	0.492	2.378
	K113	2.55	0.4	0.5	0.51	0	0.51	3.57	0.6885	2.8815
	K114	2.05	0.4	0.5	0.41	0	0.41	2.87	0.492	2.378
	K115	2.55	0.4	0.5	0.51	0	0.51	3.57	0.6885	2.8815
	K116	1.85	0.4	0.5	0.37	0	0.37	2.59	0.222	2.368
	K117	2.35	0.4	0.5	0.47	0	0.47	3.29	0.282	3.008
Column	ID	Width	Length	Thick.	Gross Quantity	Void	Quantity	Gross Quantity	Void	Quantity
	S101	0.5	0.5	3	0.75	0	0.75	6	0.372	5.628
	S102	0.8	0.4	3	0.96	0	0.96	7.2	0.574	6.626
	S103	0.8	0.4	3	0.96	0	0.96	7.2	0.574	6.626
	S104	0.5	0.5	3	0.75	0	0.75	6	0.372	5.628
	S105	0.4	0.8	3	0.96	0	0.96	7.2	0.668	6.532
	S106	0.5	0.5	3	0.75	0	0.75	6	0.9255	5.0745
	S107	0.5	0.5	3	0.75	0	0.75	6	0.9255	5.0745
	S108	0.4	0.8	3	0.96	0	0.96	7.2	0.686	6.514
	S109	0.5	0.5	3	0.75	0	0.75	6	0.372	5.628
	S110	0.8	0.4	3	0.96	0	0.96	7.2	0.568	6.632
	S111	0.8	0.4	3	0.96	0	0.96	7.2	0.568	6.632
	S112	0.5	0.5	3	0.75	0	0.75	6	0.372	5.628
							Σ23.9308			Σ165.7627

Table 3. Quantity take-off values of concrete and formwork.

QTO	Location	Exact Result	Developed BIM Software	P1	P2
Concrete (m^3)	Entrance Floor	22.9505	22.813	22.95	22.951
	First Floor	23.9308	24.07	23.79	23.939
	Total	46.8813	46.883	46.74	46.89
Formwork (m^2)	Entrance Floor	157.4335	156.811	164.23	157.75
	First Floor	165.7627	165.877	171.23	164.65
	Total	323.1962	322.688	335.46	322.4

Table 4. Percentage error values of the quantity take-off values

	Percentage Error (%)		
Item	Developed BIM Software	P1	P2
Concrete	0.004%	-0.301%	0.019%
Formwork	-0.157%	3.795%	-0.246%

The results of the comparison are given in Table 4. The comparison reveals that the developed BIM software has superior accuracy at the computation of quantity take-off of formwork and concrete. Percentage errors given in Table 4 represent that the developed BIM software has very diminutive error ranges which can be ignored completely. The developed software can be used for the quantity take-off and the determination of progress payment amounts.

The developed BIM software can process complex shapes such as half-circle, ellipses, and trapezoids. Architects tend to design extreme buildings with complex shapes. Therefore, the developed BIM software can be implemented for the design and quantity take-off process of the mentioned buildings. The state-of-the-art local quantity take-off software cannot successfully process complex shapes. The developed software has important superiorities in terms of geometric computations of complex voids.

Besides computation of quantity take-off, the developed BIM software has significant potential of 3D visualization of the building. The Ursina engine developed for game programming has powerful graphical tools for the 3D rendering of the objects. The utilization of Ursina engine provided the opportunity for 3D visualization of the 2D drawings by defining the floor height and the thicknesses of slabs, and beams. The 3D visualization provides important opportunities for architects, and engineers to realize design errors.

Developers of this software aim to produce BIM software with complete BIM features in the long term. However, the present state of the software can execute quantity take-off and 3D visualization of some of the construction items. The development team is still working on the software to enhance its capabilities.

In the case study, a reinforced concrete structure is examined. The software cannot prepare the quantity take-off of steel components such as bolts, nuts, rivet, angle and etc. However, developers aim to add a job description of steel structures and the necessary components which will enable the preparation of quantity take-off of steel structures.

The "Ursina" library is utilized as a 3D renderer which is a free open source game engine. Thanks to this library, editing operations can be carried out on 3D models. In this study, only 3D visualization is emphasized instead of 3D modeling. 3D editing can be accomplished by using the function of Ursina game engine. In this study, no 3D editing is executed but 3D editing will be executed as future work.

5. Conclusion

In this study, new BIM software which can visualize the building in 3D and perform quantity take-off of concrete and formwork is developed. The 3D visualization property enables the designer to walk inside the drawn building. The quantity take-off results are compared with two local software and the comparison revealed that the developed BIM software provides more accurate quantity take-off results. The percentage error of the quantity take-off of the developed software is within a negligible margin. However, the aforementioned small error should be eliminated by further study. The present state of the developed software can prepare quantity take-off of concrete and formwork. Scaffolding for the formwork, rebar, wall, plastering, floor covering, painting, and door & window frames will be included in the quantity take-off computations. The abovementioned quantity take-off items will be added to the software for further study.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest concerning the research, authorship, and/or publication of this article.

References

[1] Kalfa SM (2018) Building information modeling (BIM) systems and their applications in Turkey. Journal of Construction Engineering Management & Innovation 1(1): 55-66.

[2] Omaran SM, Alghamdi AA, Alharishawi SC, Hains DB. Integrating BIM and game engine for simulation interactive life cycle analysis visualization. ASCE International Conference on Computing in Civil Engineering, 17–19 June 2019, Atlanta, Georgia, USA.

[3] Delgado JMD, Oyedele L, Demian P, aBeach T (2020) A research agenda for augmented and virtual reality in architecture, engineering and construction. Advanced Engineering Informatics 45:101-122. https://doi.org/10.1016/j.aei.2020.101122.

[4] Ozcan-Deniz G (2019) Expanding applications of virtual reality in construction industry: A multiple case study approach. Journal of Construction Engineering Management & Innovation 2(2):48-66.

[5] Sarı R, Pekeriçli MK (2020) An investigation of comparison and evaluation of official BIM documents released in the USA, UK and Turkey. Journal of Construction Engineering Management & Innovation 3(1):67-84.

[6] Temel BA, Basaga HB (2020) Investigation of IFC file format for BIM based automated code compliance checking. Journal of Construction Engineering, Management & Innovation 3(2):113-130.

[7] Dhalmahapatra K, Maiti J, Krishna OB (2021) Assessment of virtual reality-based safety training simulator for electric overhead crane operations. Safety Science 139:105241.

[8] Albahbah M, Kıvrak S, Arslan G (2021) Application areas of augmented reality and virtual reality in construction project management: A scoping review. Journal of Construction Engineering, Management & Innovation 4(3):151-172.

[9] Nadeem A, Wong AK, Wong FK (2015) Bill of quantities with 3D views using building information modeling. Arabian Journal for Science and Engineering 40(9):2465-2477.

[10] Sacks R, Barak R (2007) Impact of three-dimensional parametric modeling of buildings on productivity in structural engineering practice. Automation in Construction 17:439–449.

[11] Millman KJ, Aivazis M (2011) Python for scientists and engineers. Computing in Science & Engineering 13(2):9-12.

[12] Quraishi A, Dhapekar NK (2021) Applicability of Python in civil engineering. International Research Journal of Engineering and Technology 8(1):554-556.

[13] Sarvade SM, Pore SM. Use of Python programming for interactive design of reinforced concrete structures. National Conference on Exploring New Dimensions in Teaching Learning for Quality Education, July 2019, Maharashtra, India.

[14] Bettemir ÖH (2018) Development of spreadsheet based quantity take-off and cost estimation application. Journal of Construction Engineering, Management & Innovation 1(3):108-117.

[15] Khosakitchalert C, Yabuki N, Fukuda T. The accuracy enhancement of architectural walls quantity takeoff for schematic BIM models. International Symposium on Automation and Robotics in Construction, 20-25 July 2018, Berlin, Germany.

[16] Khosakitchalert C, Yabuki N, Fukuda T., (2019a). BIM-based wall framing calculation algorithms for detailed quantity takeoff. 4th International Conference on Civil and Building Engineering

Informatics, 7-8 November 2019, Sendai, Miyagi, Japan.

[17] Khosakitchalert C, Nobuyoshi Y, Tomohiro F. Automatic concrete formwork quantıty takeoff using building information modeling. 19th International Conference on Construction Applications of Virtual Reality, 13–15 November 2019, Bangkok, Thailand.

[18] Khosakitchalert C, Yabuki N, Fukuda T (2019) Improving the accuracy of BIM-based quantity takeoff for compound elements. Automation in Construction 106:102891.

[19] Lee SK, Kim KR, Yu JH (2014) BIM and ontology-based approach for building cost estimation. Automation in Construction 41:96-105.

[20] Choi J, Kim H, Kim I (2015) Open BIM-based quantity take-off system for schematic estimation of building frame in early design stage. Journal of Computational Design and Engineering 2(1):16-25.

[21] Liu H, Lu M, Al-Hussein M (2016) Ontology-based semantic approach for construction-oriented quantity take-off from BIM models in the light-frame building industry. Advanced Engineering Informatics 30(2):190-207.

[22] Olsen D, Taylor JM (2017) Quantity take-off using building information modeling (BIM), and its limiting factors. Procedia Engineering 196:1098-1105.

[23] Kim S, Chin S, Kwon S (2019) A discrepancy analysis of BIM-based quantity take-off for building interior components. Journal of Management in Engineering 35(3):05019001.

[24] Liu H, Cheng JC, Gan VJ, Zhou S (2022) A knowledge model-based BIM framework for automatic code-compliant quantity take-off. Automation in Construction 133:104024.

[25] Bettemir ÖH, Özdemir E, Sarıcı DE (2021) Selection of construction machine by considering time-cost trade-off problem. Journal of Construction Engineering, Management & Innovation 4(3):173-186.