INDIAN INSTITUTE OF MANAGEMENT KOZHIKODE

Working Paper

IIMK/WPS/524/ITS/2022/08

March 2022

A survey of Low-Code/No-Code software development tools with an application

Lija Chandran T V[1]
Mohammed Shahid Abdulla[2]

[1] Independent Researcher, Information Technology and Systems, Email: lijachandran@gmail.com

[2] Associate Professor, Information Technology and Systems, Indian Institute of Management, Kozhikode, IIMK Campus PO, Kunnamangalam, Kozhikode, Kerala 673570, India; Email: shahid@iimk.ac.in, Phone Number (+91) 495 – 2809254

A survey of Low-Code/No-Code software development tools with an application

Abstract

The former CEO of a widely used source-code repository and version control platform said that the "Future of coding is, no coding at all". There exists a niche for software applications to be developed without any coding, i.e. no writing of source code or computer programs. This working paper focuses on the platforms which enable developing software applications with less or an absent amount of coding; with the term for these platforms in industry being Low-Code and No-Code development platforms, respectively. There has been a surge in the adoption of no-code/low-code development platforms during the COVID19 pandemic as the remote operating mode of work needed the digitization of many day-to-day physical processes and workflows. Low-code/no-code platforms are known for the convenience they offer to their customers with its easily usable drag-drop features to develop the application. Thus the applications can be developed by people who don't have programming skills, many of whom can then develop expertise as "citizen" developers for future social enterprise projects. This paper explains in detail the differences between the low-code and no-code development platforms, common use cases and the benefits over the traditional coding method. In the last section, we have designed and demonstrated a low-code application for a classroom use-case and explained the wireframes that we encountered.

1. Introduction

The way in which the software is being developed is changing. To develop a software, certain skill sets are required; a developer should be proficient in the concerned programming language/scripting language/application programming interfaces and also they should have knowledge on handling backend databases. Usually an organization will have a team of software developers to build an application. Each functionality or feature will be shared and developed by multiple developers based on their skill sets. But now with the introduction of low-code/no-code development platforms, the necessity to have a skilled developer is insignificant. Contrary to the traditional coding platforms where a developer has to write the code line by line, low-code/no-code development platforms provide pre-coded components, modules and templates which enables even the non-developers aka citizen developers to simply drag and drop the components to create an application. These platforms enable anyone to create an application with very little coding[1].

Though low-code and no-code terminologies are used together/interchangeably, no-code is a subsection of low-code. While low-code development platforms can be used to create complex applications, no-code development platforms can be used to create reporting, analytics and tracking apps[2]. *AppExchange* from Salesforce and *Zoho* are examples of low-code development platforms and *AppSheet* from Google, *Quickbase* and *Bubble* are examples of no-code development platforms[3].

Report from a world's leading research and advisory company shows that there is an increase of 22.6% in the overall revenue from low-code development platforms in 2021 compared to 2020. The following table lists the revenue from the low-code development technologies (in Million US Dollars) over the past three years from the same report[4]:

Year	2019	2020	2021
Low-Code Application Platforms	$ 3,473.50	$ 4,448.20	$ 5,751.60
Intelligent Business Process Management Suites	$ 2,509.70	$ 2,694.90	$ 2,891.60
Multi Experience Development Platforms	$ 1,583.50	$ 1,931.00	$ 2,326.90
Robotic Process Automation	$ 1,184.50	$ 1,686.00	$ 2,187.40
Citizen Automation and Development Platform	$ 341.8	$ 438.7	$ 579.5
Other Low-Code Development Technologies	$ 59.6	$ 73.4	$ 87.3
Overall Revenue	$ 9,152.6	$ 11,272.2	$ 13,824.2

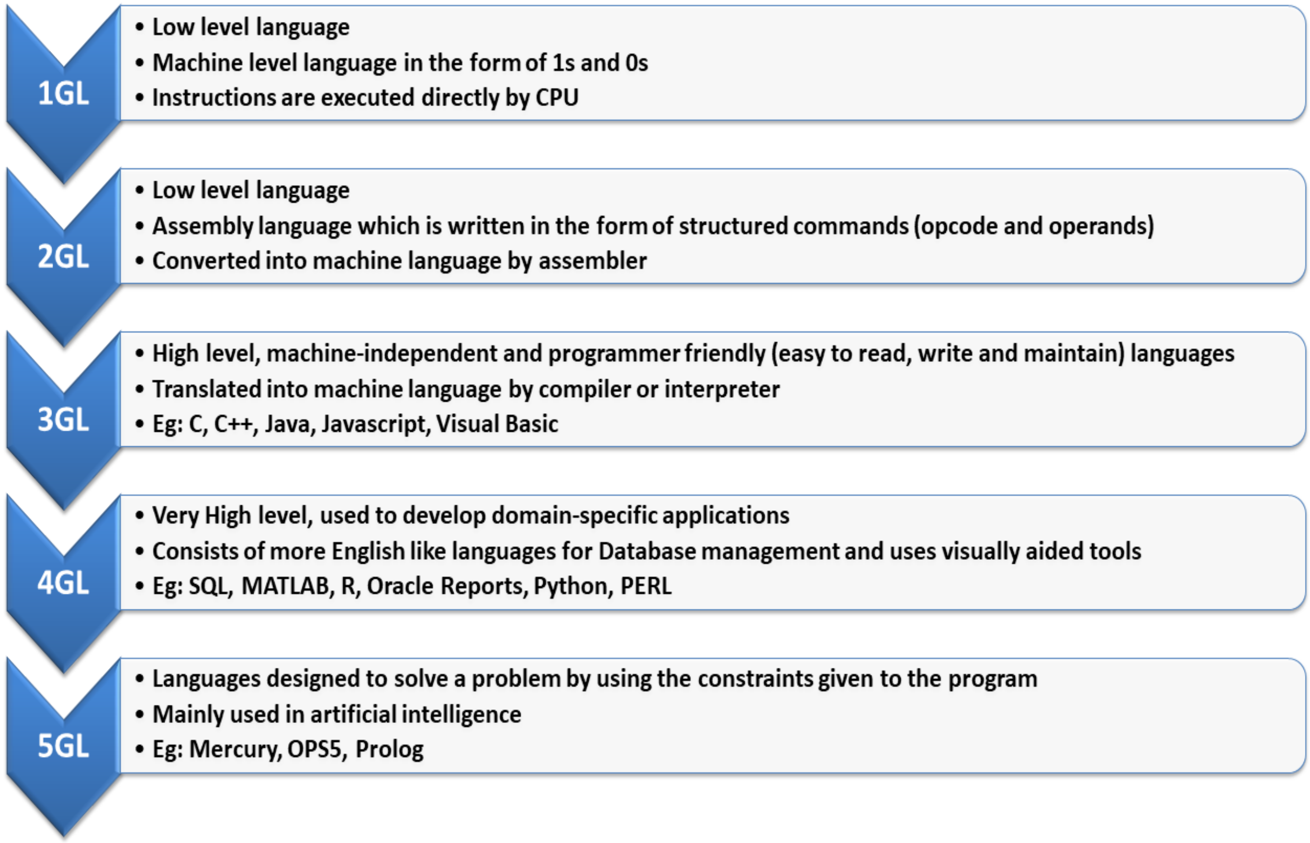
Table: 1

Another report says that 44% of the software development in 2020 was done using low-code development platforms and by the end of 2021 it is estimated that 75% of the software development will be done using low-code development platforms[5].

2. Generations of Programming Language

There are five generations of programming languages starting from the machine language which is called the first generation programming language (1GL)[6]. Usage of visually aided programming platforms like *Visual Basic* started from the third generation of programming languages[7]. Fourth generation of programming languages consisted of codeless programming tools like table-driven programming and report-generators. The fourth generation of programming languages emerged as the low-code programming platforms[8].

The following figure shows the evolution of programming languages.



1GL

- Low level language
- Machine level language in the form of 1s and 0s
- Instructions are executed directly by CPU

2GL

- Low level language
- Assembly language which is written in the form of structured commands (opcode and operands)
- Converted into machine language by assembler

3GL

- High level, machine-independent and programmer friendly (easy to read, write and maintain) languages
- Translated into machine language by compiler or interpreter
- Eg: C, C++, Java, Javascript, Visual Basic

4GL

- Very High level, used to develop domain-specific applications
- Consists of more English like languages for Database management and uses visually aided tools
- Eg: SQL, MATLAB, R, Oracle Reports, Python, PERL

5GL

- Languages designed to solve a problem by using the constraints given to the program
- Mainly used in artificial intelligence
- Eg: Mercury, OPS5, Prolog

Figure : 1

Fifth generation programming language is used to develop applications that can solve problems based on the constraints. It enables the machines to learn to solve problems and make decisions, i.e. to create artificial intelligence[9].

3. Adoption of Low-code/No-Code platforms

Most of the product-based and service-based companies needed to create simple digital applications in order to connect with their employees, clients and customers during the Covid-19 pandemic. Digitization of many of the use cases had to be done as quickly as possible to retain the day-to-day processes of the organizations unaffected. The applications which would have taken nine to twelve months for development using the traditional coding method can now be developed within a few weeks using low-code/no-code platforms. This led to a surge in the adoption of low-code/no-code platforms to create applications rapidly[10].

An analyst firm's leader predicted that there will be an accelerated adoption of low-code platforms in 2021 because of the pandemic and there could even be a change in composition of the development team; business users and professional developers will be able to build the applications together employing such low-code platforms[11]. Following graph shows the projected low-code development platform market growth by the same analyst firm[12].

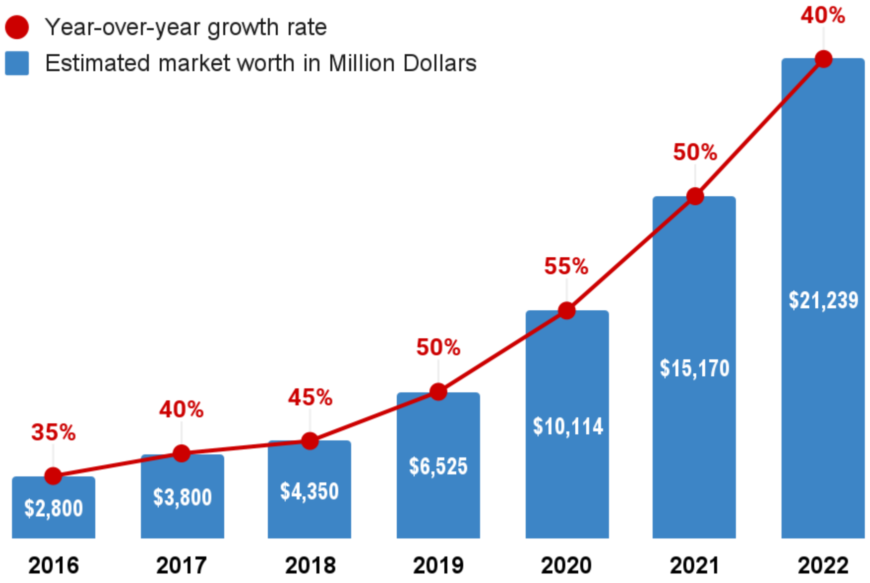


Figure : 2

With the help of a low-code platform, a customized application for the citizens to get registered for Covid-19 test was designed, built and launched in 72 hours by the Los Angeles local body[13]. A similar incident was reported in New York city, where they deployed a Covid-19 engagement portal in 72 hours, again with the help of a no-code platform[14].

4. Low-Code vs No-Code

As it can be easily inferred from the names, low-code platforms have the provision to add the custom codes when required, whereas the no-code platforms don't have such provisions. These platforms provide pre-coded templates which can be customized according to the requirements and then can be published in different formats suited for desktops/laptops, tablets and mobile phones. Low-code platforms provide an additional facility to add custom logic by writing our own codes, and thus need some programming skills. The language used to code might be a proprietary language which is a *wrapper* on top of the existing programming languages and this saves time by writing code only once, which will automatically be converted into the multiple programming languages required to support different kinds of devices as mentioned earlier. In a no-code platform, we can only choose from the in-built modules and events to create the application which limits us from customizing the app extensively - though many immediate purposes can be served.

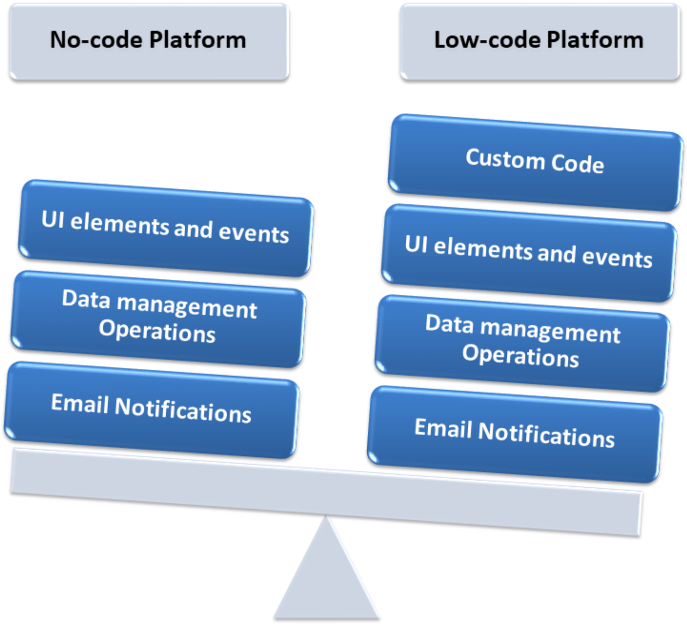


Figure : 3

Both platforms have pre-coded visual elements like textboxes, buttons, drop-downs, checkboxes etc. which can be dragged and dropped into forms or pages. The data entered into the forms are automatically added into backend tables, which again saves time of creating a database, or connecting the front-end to the back-end. Most of the applications used in an enterprise like Human Resources, Helpdesk, Finance, Sales and Marketing rely on data. Low/No-code platforms provide pre-built modules which can handle typical actions in a process flow, e.g. email notifications. These inbuilt capabilities enable the low-code no-code platforms to develop applications in a very short span as compared to the traditional coding.

5. Traditional Coding vs. Low-Code/No-Code

A survey conducted by a low-code platform provider across six countries among 1209 IT leaders and 816 software developers says that low-code projects reduce costs by 53%, can be developed 56% faster, and the customer revenue increased by an average of 58% compared to the traditionally built applications[15]. Following are some of the key differentiating factors between the traditional coding and low-code/no-code:

Skillset:
To develop an application using the traditional coding method requires professional developers who are expert in different programming languages. There might be a need for different programmers who have expertise in developing the front-end for a web-based application and mobile-app, also another set of programmers who has expertise on handling the back-end, or simply extra- qualified *'full stack'* developers for any novel system being built. In contrast, in low-code/no-code platforms both 'citizen' developers with rudimentary skills and professional developers can build an application.

Duration:
As mentioned in section 3, automating the supply chain of a business could take up to nine to twelve months using the traditional coding method as it involves extensive coding, multiple code revisions, testing and bug fixes. It might appear to readers that 9-12 months is an overestimate, when in many Cos the attachment of a software developer to a project/product is for 1 quarter, ie. 3 months. Whereas with the help of no-code/low-code platforms the same can be built within weeks. Pre-coded templates are also available to choose from.

Cost:
Here we present a comparison claimed by a no-code platform. The estimated budget to develop a ride sharing app for Android or iOS platform in the traditional way is between $55,000 to $120,000[16]. Whereas similar apps developed in this company's platform can be hosted in a custom domain and the charges start from $25 per month[17].

Deployment Platform Support:
In traditional coding web applications and mobile apps need different sets of developers with different skill sets. Thus these are designed, developed and deployed separately. But the applications built using no-code/low-code platforms can be deployed as web application, Android app and iOS app all at once.

6. Use Cases

No-code/low-code development platforms can be used to design, build and deploy a variety of applications, here are some common use cases :

➔ **Rapid Prototyping**: COO of a leading service based company from India says that the no-code/low-code platforms are very useful in demonstrating the prototypes of the applications to their clients[18]. Suggestions for low-code/no-code use cases on similar lines are prototypes in a business school's product management class, where not all students might have a software development background.

➔ **Customer Engagement Apps**: Low-code/no-code platform can be used to build the customer engagement apps like chatbots, virtual assistants, feedback forms and surveys. There are customer engagement app templates provided by these platforms which can be easily customized[19]. Of these, mobile-based feedback forms and surveys - even to actively run business processes in a firm - are currently in vogue.

➔ **AI and Multi-experience apps**: Low-code/no-code AI platforms can be used to create the multi-experience applications such as Virtual Reality and Augmented Reality. The applications like chatbots would come under this category as well because of the artificial intelligence requirements[19]. In both these use cases, there must be an existing platform where these add-on modules apply well. For example, without an e-commerce platform where past chats (or voice transcripts of customer calls) aren't available - chatbots may not be a success, esp. in dealing with customer voice or tone.

➔ **Operational Efficiency Apps**: The apps which can be used by the employees of an organization to automate the manual or paper-based processes. These kinds of apps can be built easily with the help of no-code/low-code platforms[19]. This can also include robotic process automation where the data from a legacy application has to be copied to the current application or vice-versa - or other repetitive routine tasks[20]. For instance, a low-code mobile phone app will provide employees with information on an organization's conveyance arrangement - incl. input from peripherals such as GPS.

➔ **Visual Analytics and Dashboards**: Low-code/no-code platforms can access the corporated data in real time and build real time business analytics applications and dashboard reporting tools faster than traditional coding [21].

➔ **Data Management Applications**: Customized database based applications which aid in collecting, storing, updating and sharing information can be built easily by connecting to the organization's database[21]. An example of this could be a company's sales team member's phone app, updated after each visit to an outlet stocking the company's product.

➔ **Migration of Legacy Apps**: Migrating the legacy applications to modern systems is a very time consuming and difficult task for the organization. The entire process of coding can be eliminated by using no-code/low-code platforms[21]. There are low-code platforms which offer to migrate - incl. the services required - the legacy ERP systems to a modern system in 12 months[22].

7. Points to ponder

Following are some points to be considered before choosing to develop an application using no-code/low-code platforms:

➔ Though there are many low-code/no-code development platforms available to build simple customer/employee facing apps. It is worth noting that there are alternative platforms like *Google Forms* and *Microsoft Forms* using which many of these use cases can be designed and deployed, esp. such use cases where electronic peripherals or sensors such as those present in mobile phones *aren't* being used. These freely available platforms enable the creation of different kinds of forms eg: feedback, quiz, item order, survey form, job application, time-off request, time-slot booking, polls and surveys, and a control-based workflow between multiple forms based on an encoded sequence. There is also a provision to customize the theme, background image and color of these forms[23].

➔ Use of no-code/low-code platforms in an organization can also lead to shadow IT. If the citizen developer who built the application using the no-code/low-code platform leaves the organization, it would be moderately difficult for the central IT department or another employee to modify or support the application - e.g. attach enhancements. So, it is suggested to have a hybrid development team where a professional developer is also involved in certain stages of building the application along with the citizen developer[24].

- ➔ Choosing the low-code/no-code development platforms to build applications easily and faster might also result in technical debt, i.e. poor architecture choices for the software or design patterns in the features or in the processing that can cause future problems or bottlenecks. When the applications are built by citizen developers without proper supervision, this can lead to compliance issues, integration problems and resource utilization issues etc. which pile up to cause issues in the future [25].

- ➔ It is difficult to audit the security standard implemented in the applications developed using no-code/low-code platforms as there is no access to source code[26]. The low-code/no-code application's features are translated into an underlying language by the platform's compiler. But such code is likely to have minimal readability or logic.

- ➔ Maintenance of the applications developed using no-code/low-code platforms will be challenging in the long run as there is always a dependence on these platform vendors[26], in addition to the business users who created the app, who might suffer attrition.

8. Demonstration

A custom application to create and randomize quizzes is built from scratch using a low-code platform. The purpose of the application is to create multiple sets of quiz questions by randomizing the questions. This will be useful for the academic instructors and teaching associates/academic associates where the quiz is a component of the course. When there is a need to conduct the offline quizzes regularly with multiple sets of question papers this application can help to create different sets of questions with a click of a button, as opposed to randomizing the questions manually. The following forms and tasks could be developed in one working day (~9 hrs):

1. Home screen where all the options are listed
2. Create the quiz by entering an unique quiz name and clicking on "Create" button
3. List all the quizzes that are already created
4. Add questions to the already existing quiz by selecting the quiz name, enter the quiz question, four options and select the right option and click on the "Add Question button".
5. Question Bank screen where all the questions of the quizzes will be displayed in the selected order.

6. Randomize screen where an existing quiz can be selected and click on the “Randomize button” and it will trigger an email to the instructor's mail id with one set of randomized questions.

All the screens are made using the drag-and-drop utility of the Zoho Creator, which is a software’s company’s custom low-code application development platform. In addition, the checks and conditions are also set by changing the properties of the corresponding UI elements. Steps 1 to 5 did not involve any coding. Step 6 is where custom code is written to fetch all the questions of the selected quiz, randomize the order of the quiz questions (in this demonstration logic was incomplete to create multiple sets of quizzes) and send email (using the platform’s email plugin) to the instructor.

An application like this, with added design features, can thus be developed over a few working days and is available to be deployed as a web application, Android app or iOS app by paying the appropriate license fee for Zoho.

Figures 4 and 5 are power-user’s screenshots of the mobile application version of the quiz randomization application, figures 6 to 11 are the web application view of power user, figure 12 and 13 shows the developers view of the Zoho creator and figure 14 is the source code:

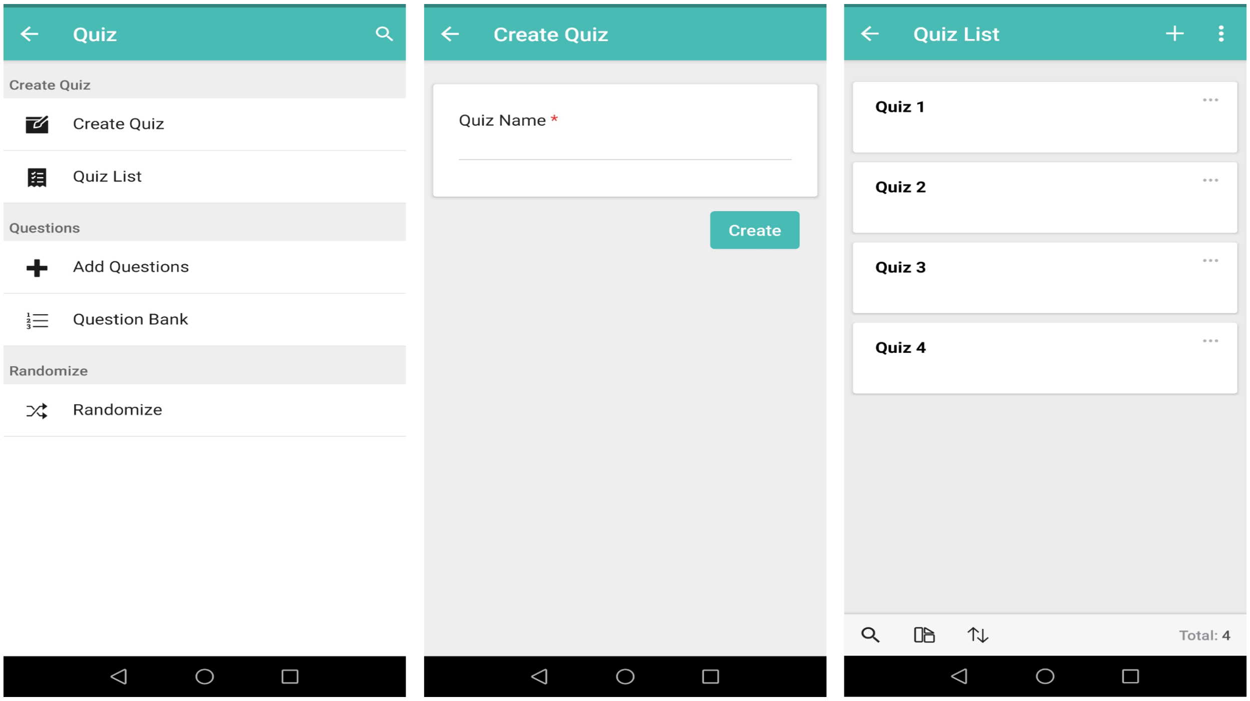
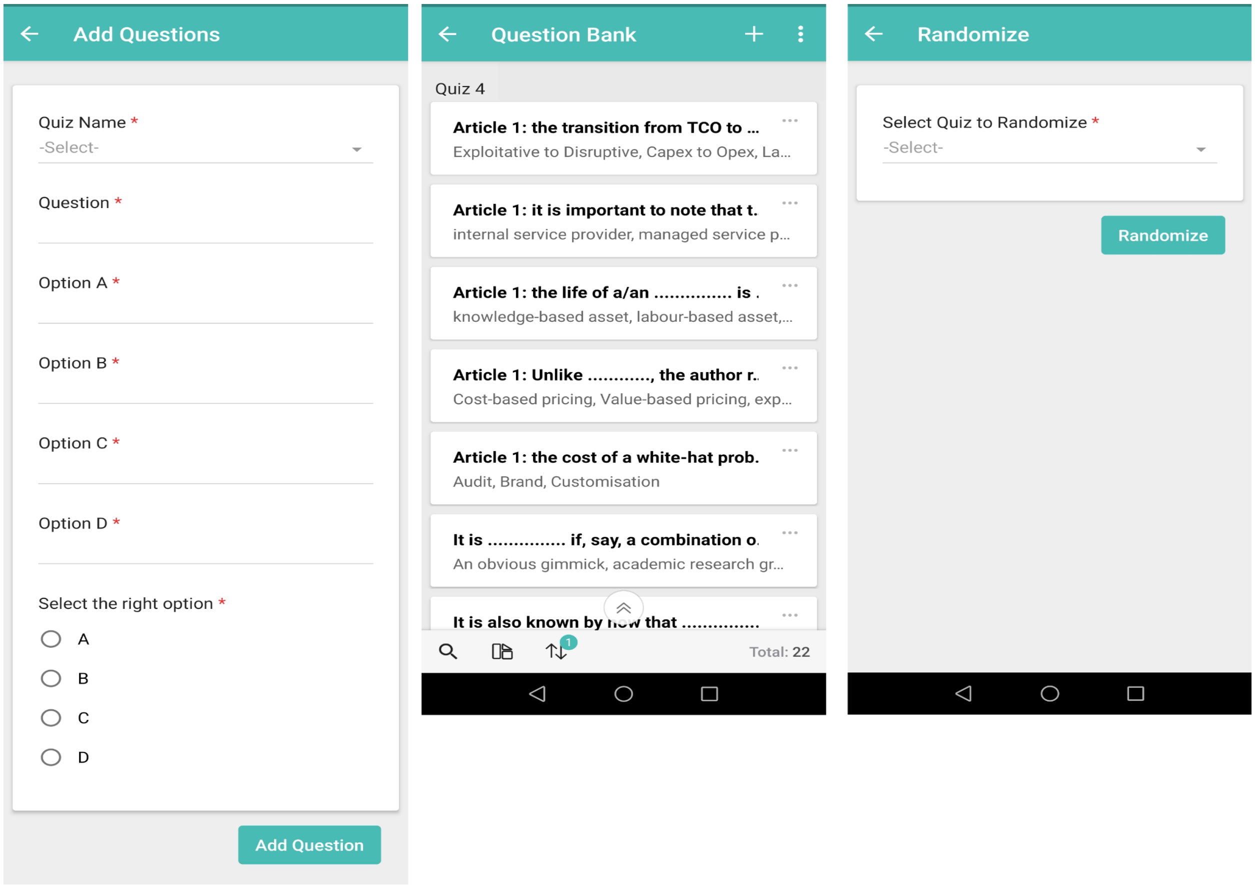


Figure 4: Mobile app screens



Quiz 4

Article 1: the transition from TCO to ...
Exploitative to Disruptive, Capex to Opex, La...

Article 1: it is important to note that t.
internal service provider, managed service p...

Article 1: the life of a/an .... is .
knowledge-based asset, labour-based asset,...

Article 1: Unlike ...., the author r..
Cost-based pricing, Value-based pricing, exp...

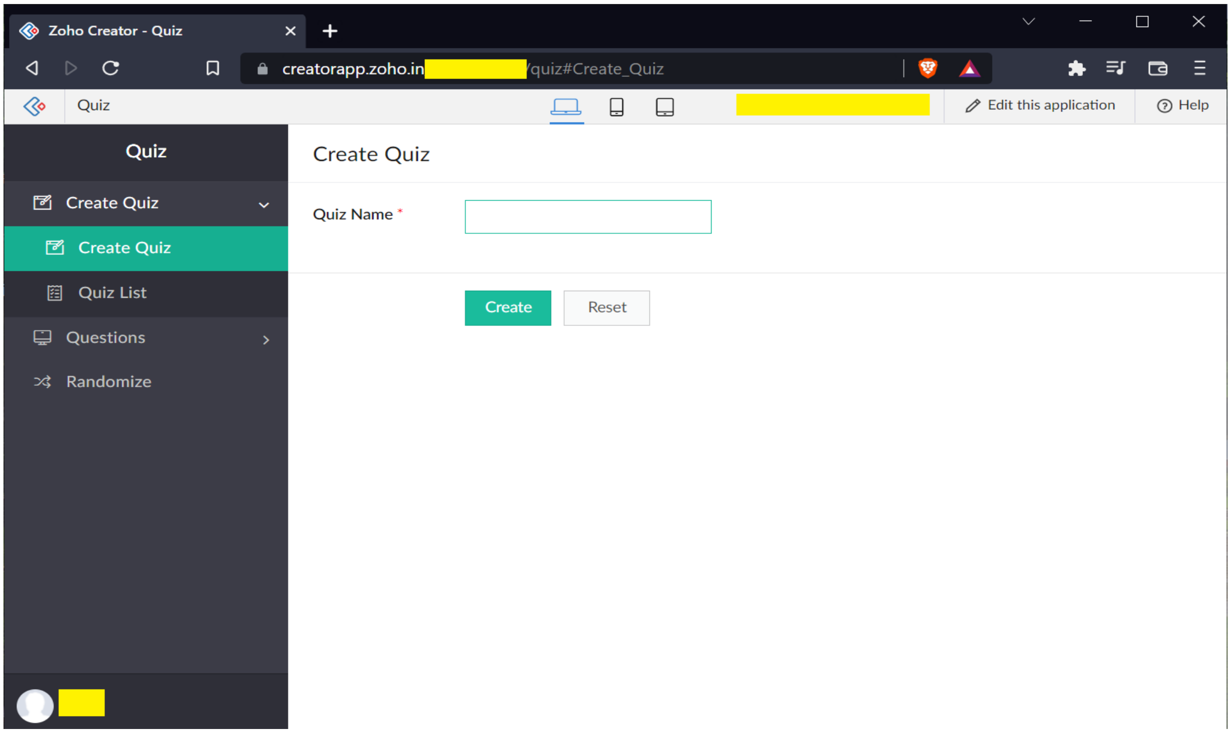
Article 1: the cost of a white-hat prob.
Audit, Brand, Customisation

It is .... if, say, a combination o.
An obvious gimmick, academic research gr...

It is also known by how that ....

Total: 22

Figure 5: Mobile app screens

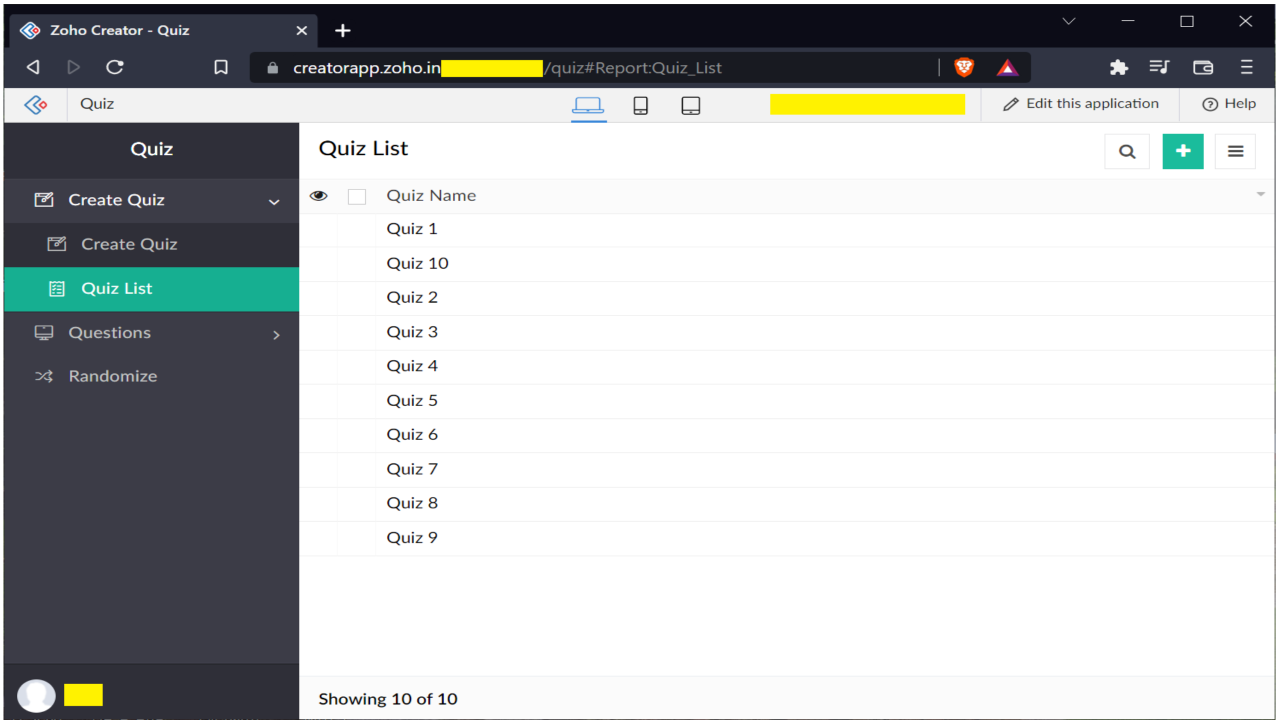


creatorapp.zoho.in[redacted]/quiz#Create_Quiz

[redacted]

[redacted]

Figure 6:Web application view of Create Quiz Page

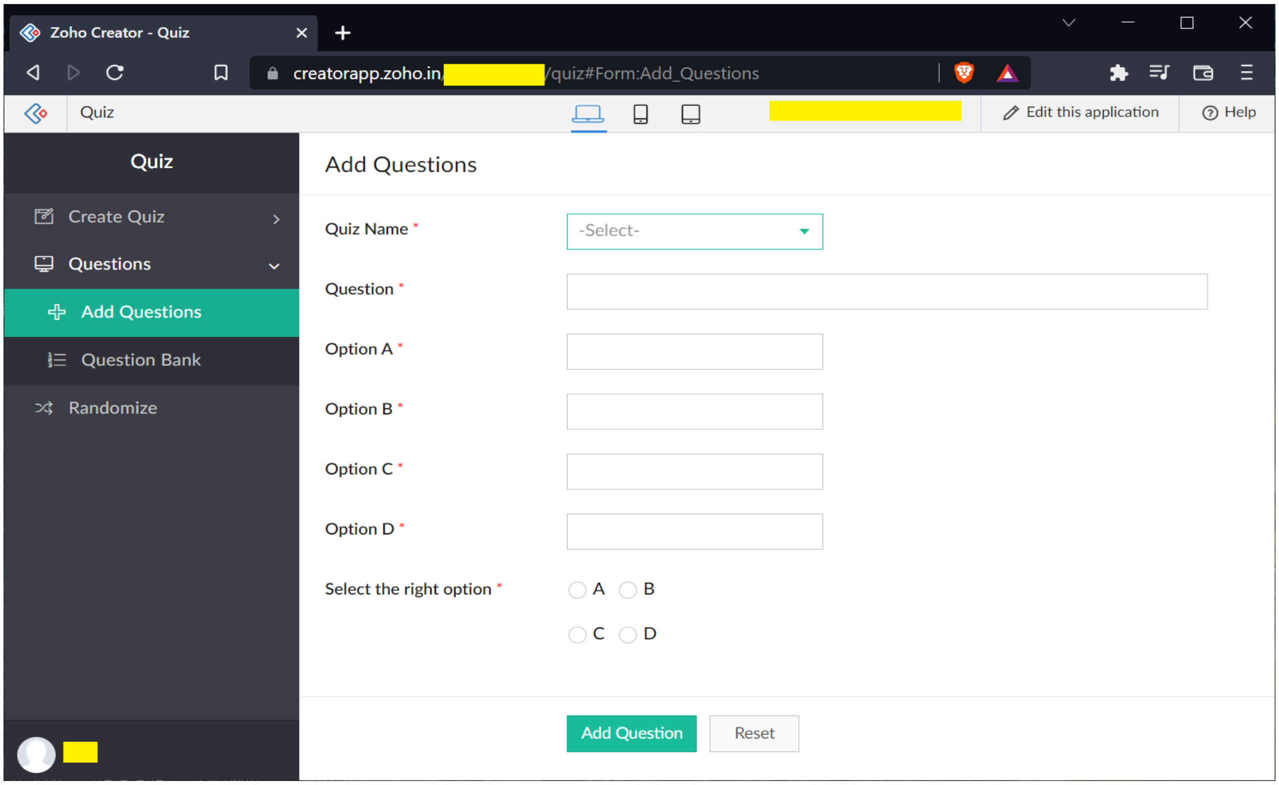


Quiz List

Quiz Name
Quiz 1
Quiz 10
Quiz 2
Quiz 3
Quiz 4
Quiz 5
Quiz 6
Quiz 7
Quiz 8
Quiz 9

Showing 10 of 10

Figure 7: Web application view of List of Quizzes



Add Questions

Quiz Name * -Select-

Question *

Option A *

Option B *

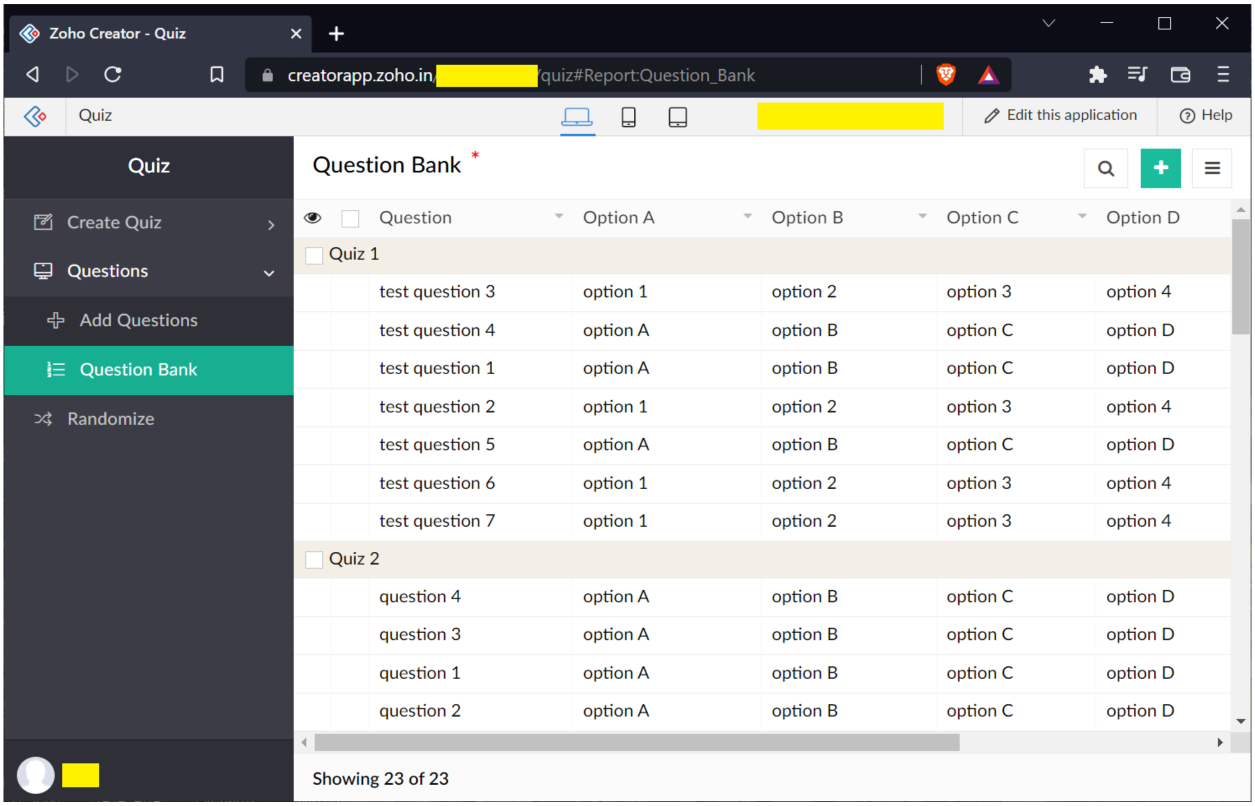
Option C *

Option D *

Select the right option * ○ A ○ B ○ C ○ D

Add Question   Reset

Figure 8: Web application view of Add Question Page

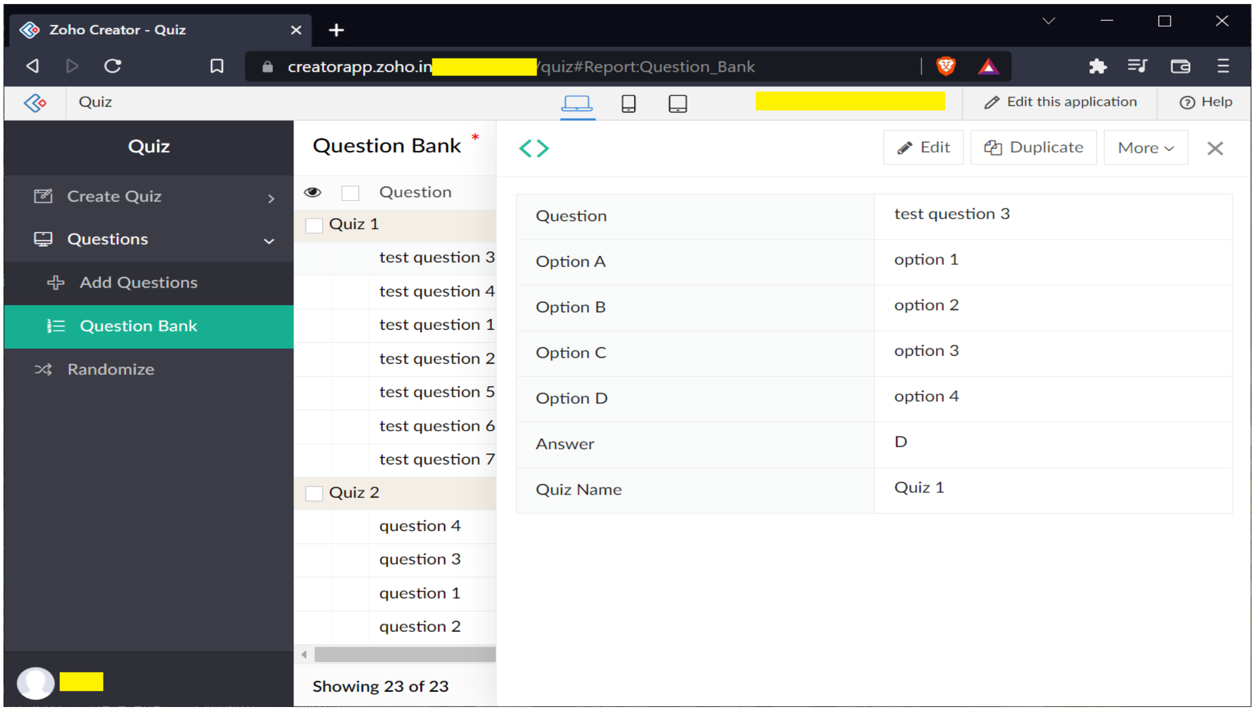


Question Bank *

Question	Option A	Option B	Option C	Option D
□ Quiz 1				
test question 3	option 1	option 2	option 3	option 4
test question 4	option A	option B	option C	option D
test question 1	option A	option B	option C	option D
test question 2	option 1	option 2	option 3	option 4
test question 5	option A	option B	option C	option D
test question 6	option 1	option 2	option 3	option 4
test question 7	option 1	option 2	option 3	option 4
□ Quiz 2				
question 4	option A	option B	option C	option D
question 3	option A	option B	option C	option D
question 1	option A	option B	option C	option D
question 2	option A	option B	option C	option D

Showing 23 of 23

Figure 9: Web application view of Question Bank



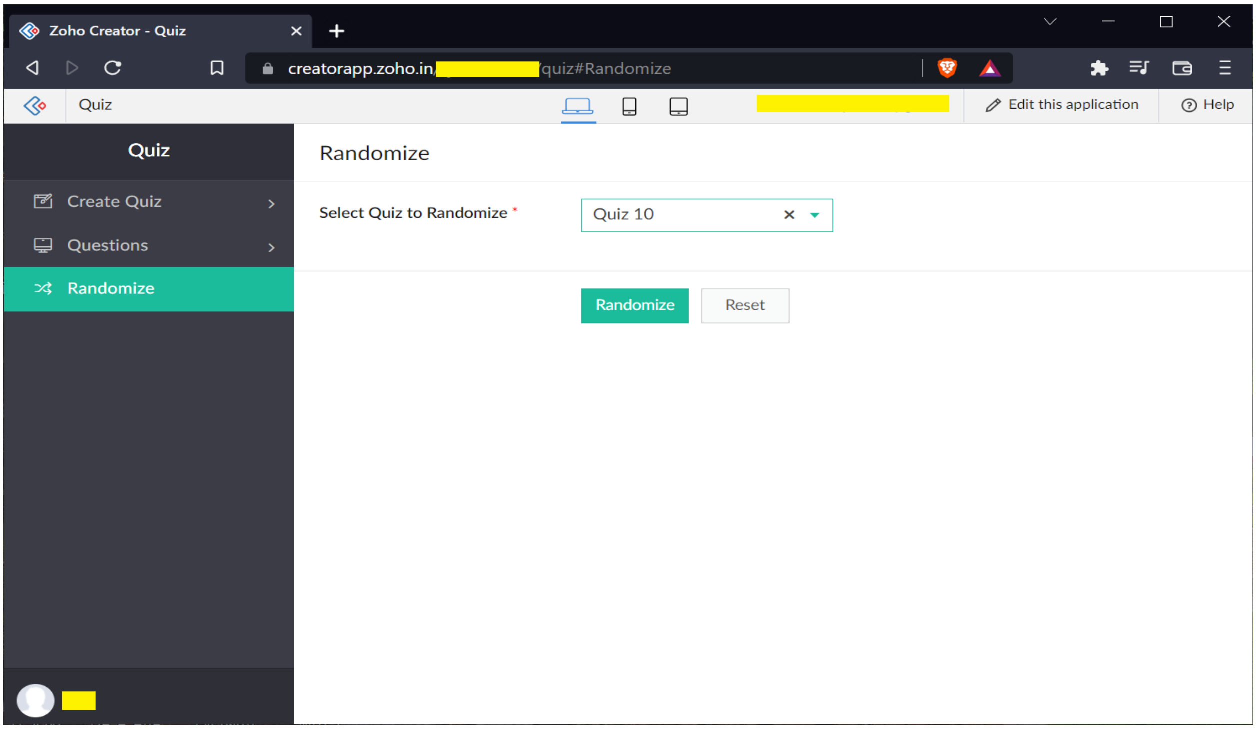
Question Bank *

Question
□ Quiz 1
test question 3
test question 4
test question 1
test question 2
test question 5
test question 6
test question 7
□ Quiz 2
question 4
question 3
question 1
question 2

Showing 23 of 23

Question	test question 3
Option A	option 1
Option B	option 2
Option C	option 3
Option D	option 4
Answer	D
Quiz Name	Quiz 1

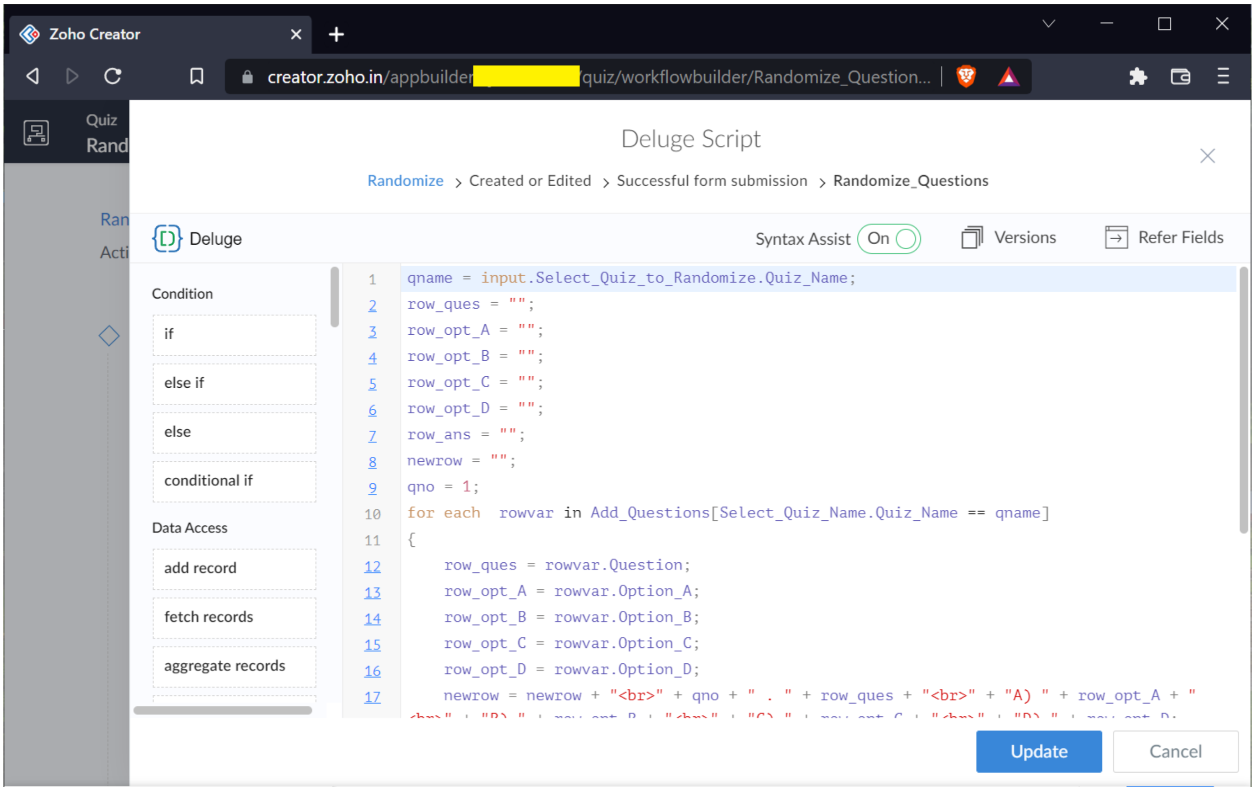
Figure 10: Web application view of Edit Questions



Randomize

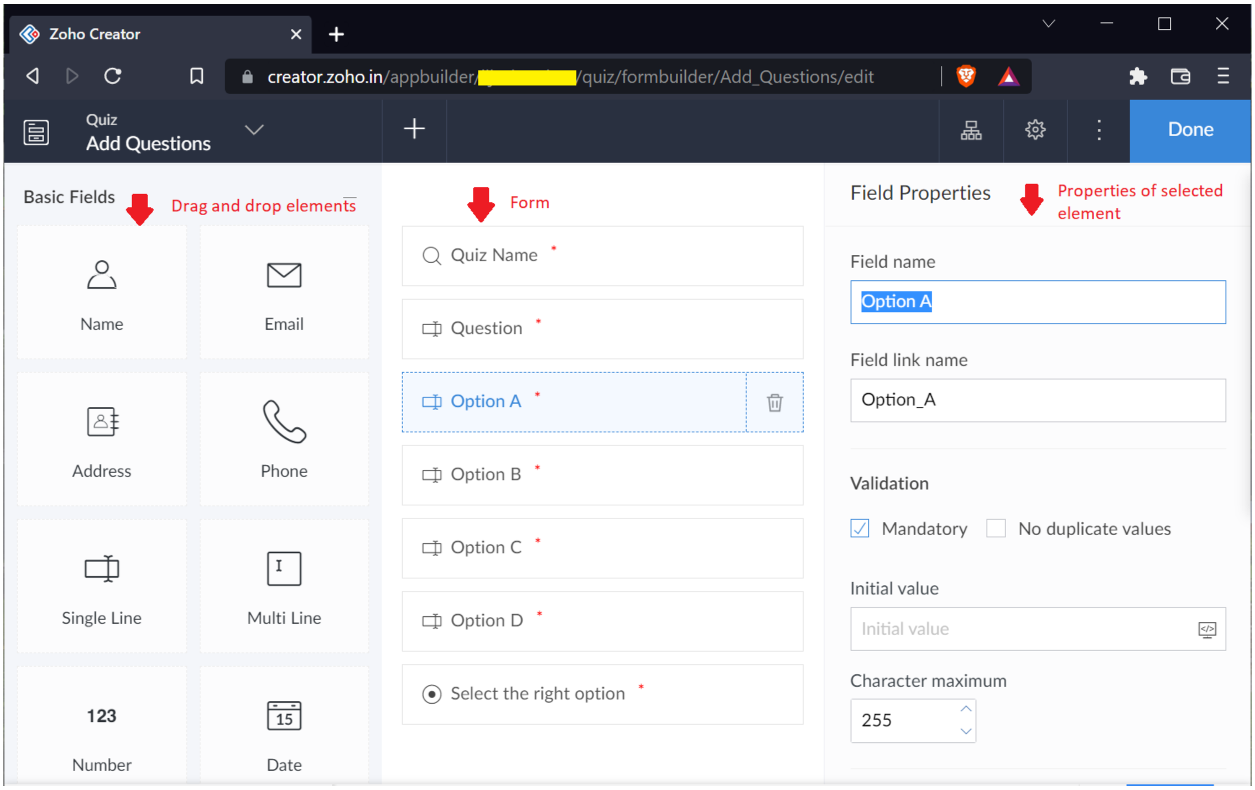
Select Quiz to Randomize * Quiz 10

Figure 11: Web application view of Randomize Quiz Page



```
qname = input.Select_Quiz_to_Randomize.Quiz_Name;
row_ques = "";
row_opt_A = "";
row_opt_B = "";
row_opt_C = "";
row_opt_D = "";
row_ans = "";
newrow = "";
qno = 1;
for each  rowvar in Add_Questions[Select_Quiz_Name.Quiz_Name == qname]
{
    row_ques = rowvar.Question;
    row_opt_A = rowvar.Option_A;
    row_opt_B = rowvar.Option_B;
    row_opt_C = rowvar.Option_C;
    row_opt_D = rowvar.Option_D;
    newrow = newrow + "<br>" + qno + " . " + row_ques + "<br>" + "A) " + row_opt_A + "
<br>" + "B) " + row_opt_B + "<br>" + "C) " + row_opt_C + "<br>" + "D) " + row_opt_D;
```

Figure 12: Developers view of coding page



Quiz Name *

Question *

Option A *

Option B *

Option C *

Option D *

○ Select the right option *

Field name: Option A

Field link name: Option_A

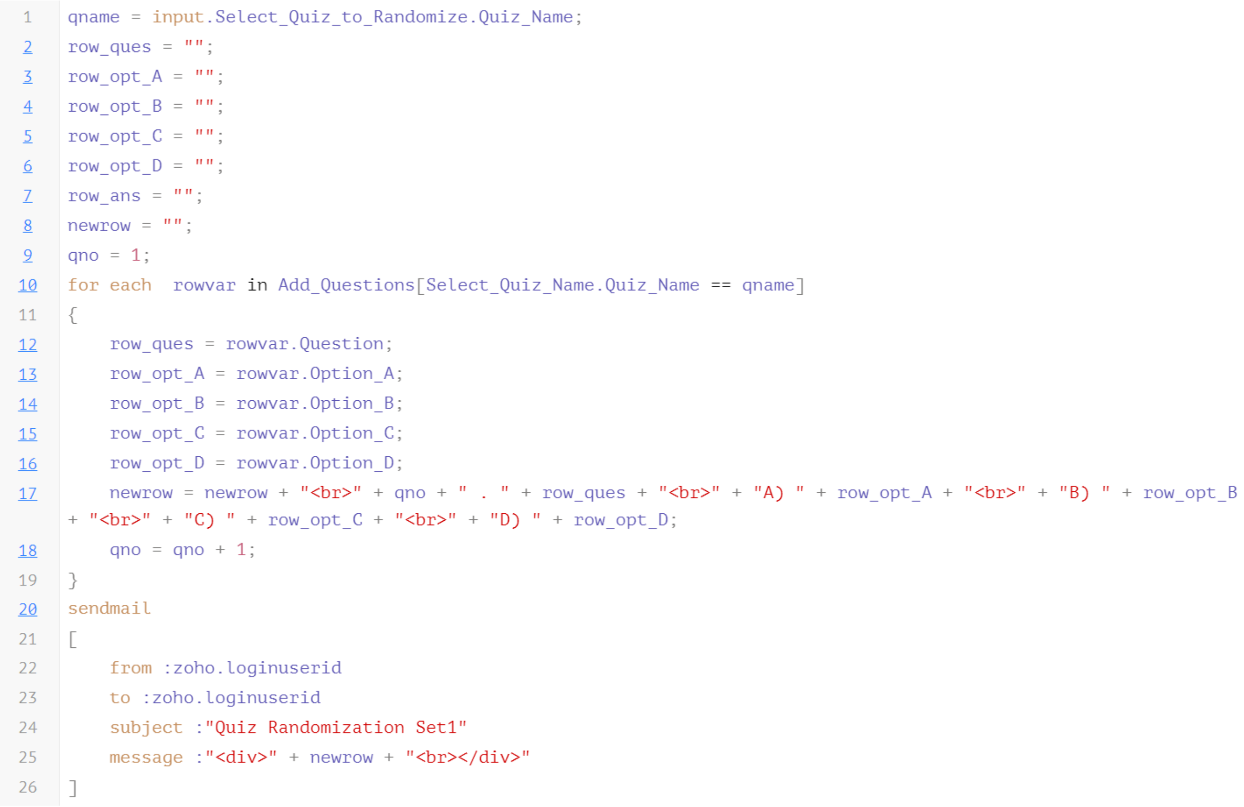
Validation

☑ Mandatory □ No duplicate values

Initial value: Initial value

Character maximum: 255

Figure 13: Developers view of form edit page



```
qname = input.Select_Quiz_to_Randomize.Quiz_Name;
row_ques = "";
row_opt_A = "";
row_opt_B = "";
row_opt_C = "";
row_opt_D = "";
row_ans = "";
newrow = "";
qno = 1;
for each  rowvar in Add_Questions[Select_Quiz_Name.Quiz_Name == qname]
{
    row_ques = rowvar.Question;
    row_opt_A = rowvar.Option_A;
    row_opt_B = rowvar.Option_B;
    row_opt_C = rowvar.Option_C;
    row_opt_D = rowvar.Option_D;
    newrow = newrow + "<br>" + qno + " . " + row_ques + "<br>" + "A) " + row_opt_A + "<br>" + "B) " + row_opt_B + "<br>" + "C) " + row_opt_C + "<br>" + "D) " + row_opt_D;
    qno = qno + 1;
}
sendmail
[
    from :zoho.loginuserid
    to :zoho.loginuserid
    subject :"Quiz Randomization Set1"
    message :"<div>" + newrow + "<br></div>"
]
```

Figure 14: Source Code

9. Conclusion

The no-code development platforms can be used by citizen developers to create software applications with no amount of coding and the low-code development platforms can be used by people who have some knowledge in coding with the lowest possible amount of coding. These platforms can be used to develop and deploy applications faster, with reduced cost and increased revenue, compared to the traditional coding methodology.

References

[1] *Low-code development platform*. (2017, January 25). Wikipedia. https://en.wikipedia.org/wiki/Low-code_development_platform

[2] Miller, A. (2021, July 30). *Low Code vs No Code Explained*. BMC Blogs. https://www.bmc.com/blogs/low-code-vs-no-code/#

[3] *No-code development platform*. (2018, August 20). Wikipedia. https://en.wikipedia.org/wiki/No-code_development_platform

[4] *Gartner Forecasts Worldwide Low-Code Development Technologies Market to Grow 23% in 2021*. (2021, February 16). Gartner. https://www.gartner.com/en/newsroom/press-releases/2021-02-15-gartner-forecasts-worldwide-low-code-development-technologies-market-to-grow-23-percent-in-2021

[5] *Forrester predicts significant increase in low-code platform adoption in 2021 & beyond*. (2020, December 9). Appian. https://appian.com/blog/2020/forrester-predicts-significant-increase-in-low-code-platform-adoption-in-2021-andamp-beyond.html

[6] *Programming language generations*. (2009, September 23). Wikipedia. https://en.wikipedia.org/wiki/Programming_language_generations

[7] *Visual Basic (classic)*. (2001, October 7). Wikipedia. https://en.wikipedia.org/wiki/Visual_Basic_(classic)

[8] *Fourth-generation programming language*. (2001, October 15). Wikipedia. https://en.wikipedia.org/wiki/Fourth-generation_programming_language

[9] *Generation of programming languages*. (2021, August 29). GeeksforGeeks. https://www.geeksforgeeks.org/generation-programming-languages/

[10] *Low-code no-code boom during COVID-19 pandemic*. (2020, November 11). NASSCOM Community. https://community.nasscom.in/communities/emerging-tech/low-code-no-code-boom-during-covid-19-pandemic.html

[11] Hammond, J. (2020, October 30). *Predictions 2021: Software developers face mounting pressure*. Forrester. https://www.forrester.com/blogs/predictions-2021-software-developers-face-mounting-pressure/

[12] Rymer, J. (2018, September 21). *Why you need to know about low-code, even if you're not responsible for software delivery*. Forrester. https://www.forrester.com/blogs/why-you-need-to-know-about-low-code-even-if-youre-not-responsible-for-software-delivery/

[13] *City of Los Angeles – ServiceNow – Customer story*. (accessed in October 2021). ServiceNow. https://www.servicenow.com/customers/los-angeles.html

[14] Unqork. (2020, April 1). *New York City has deployed a COVID-19 engagement Portal built in partnership with Unqork to manage growing crisis*. PR Newswire. https://www.prnewswire.com/news-releases/new-york-city-has-deployed-a-covid-19-engagement-portal-built-in-partnership-with-unqork-to-manage-growing-crisis-301033065.html

[15] *THE STATE OF LOW-CODE 2021: A LOOK BACK, THE LIGHT AHEAD*. (2021). Mendix. https://www.mendix.com/wp-content/uploads/Mendix_2021_State_of_LowCode_Whitepaper.pdf

[16] Martin, S. (2021, June 10). *Develop An App Like Uber in 2021 (Cost, Features, Business Model etc.)*. Medium. https://medium.com/flutter-community/how-to-develop-an-app-like-uber-for-android-ios-cost-feature-and-more-5649e2ee1c93

[17] Kirkwood, L. (2020, September 11). *How To Build An Uber Clone Without Writing Code*. Bubble Blog. https://bubble.io/blog/build-uber-clone-no-code/

[18] *Indian IT firms take to low-code, no-code solutions*. (2021, February 22). The Economic Times. https://economictimes.indiatimes.com/tech/information-tech/indian-it-firms-take-to-low-code-no-code-solutions/articleshow/81144108.cms

[19] *Low code use cases: What can you build with low code platforms*. (2021, September 6). Gramener Blog. https://blog.gramener.com/low-code-platform-use-cases/

[20] Siciliano, S. (2021, October 8). *Automation in Windows 11 enables users to reduce repetitive tasks and boost productivity*. Microsoft Power Platform Blog. https://cloudblogs.microsoft.com/powerplatform/2021/10/04/automation-in-windows-11-enables-users-to-reduce-repetitive-tasks-and-boost-productivity/

[21] Balo, S. (2020, February 6). *5 best use cases of no code*. Medium. https://medium.com/@senolbalo/5-best-use-cases-of-no-code-5141e362b329

[22] Evolution of the ERP system. (2020, July 25). fizor. https://fizor.io/services/low-code-no-code-erp-systems/

[23] Bisson, S. (2020, December 15). *Microsoft vs Google: Forms apps for surveys, polls, and quizzes*. TechRepublic. https://www.techrepublic.com/article/microsoft-vs-google-forms-apps-for-surveys-polls-and-quizzes/

[24] Johannessen, C., & Davenport, T. (2021, June 22). *When low-code/no-code development works — and when it doesn't*. Harvard Business Review. https://hbr.org/2021/06/when-low-code-no-code-development-works-and-when-it-doesnt

[25] Alexander, F. (2021, January 8). *Low-code and no-code: What's the difference and when to use what?* OutSystems. https://www.outsystems.com/blog/posts/low-code-vs-no-code/

[26] Tozzi, C. (2021, March 1). *A practical take on low-code vs. traditional development*. SearchSoftwareQuality. https://searchsoftwarequality.techtarget.com/tip/A-practical-take-on-low-code-vs-traditional-development

Research Office
Indian Institute of Management Kozhikode
IIMK Campus P. O.,
Kozhikode, Kerala, India,
PIN - 673 570
Phone: +91-495-2809237/ 238
Email: research@iimk.ac.in
Web: https://iimk.ac.in/faculty/publicationmenu.php