



ARTIFICIAL INTELLIGENCE-GUIDED WEB DEVELOPMENT - GENERATING MONGODB QUERIES

Tanja Krunić*

[0009-0007-4352-3433]

The Higher Education Technical School
of Professional Studies in Novi Sad,
Novi Sad, Serbia

Abstract:

The topic of this paper is artificial intelligence (AI) guided web development with a focus on MongoDB databases. An online tool for automated generating code is used to investigate the issues arising in artificial intelligence-guided database query writing. For that purpose, a database in the MongoDB Compass framework is created. Within the research, a set of 28 prompts for generating appropriate database queries has been given to the tool. Then all obtained MongoDB queries are tested to see if they are correct by implementing them directly on the database in MongoDB Compass. All incorrect queries are analyzed, and a more precise prompt to the code generator tool is given subsequently. This process is repeated until a correct query is obtained. The aim has been to find out how precise a prompt to AI should be and what we should pay attention to if we want to obtain correct queries.

Keywords:

Artificial intelligence tools, Code generator tools, Web development, MongoDB.

INTRODUCTION

Artificial intelligence (AI) is a technology that enables computer systems to simulate human intelligence including speech and pattern recognition, natural language processing, decision-making, etc. [1]. Artificial intelligence is also able to solve problems [2] and to take actions to reach some predefined targets, [3]. Although the interest in AI research started in the second half of the 20th century, the main progress has been made in the past five years, [4]. AI is present and has a growing future in all segments of life like banking, finance, robotics, and health-care, [5], and will impact jobs in the following areas: writing, office assistance, programming, customer services, legal jobs, teaching, finance, engineering, human resources, [6]. Nowadays, there are numerous AI tools for code generation, which is a reason why some programmers fear that AI might take over their jobs. It is well known that every fear can be diminished by getting familiar with the 'enemy' and trying to exploit it. In other words, programmers should learn how to use AI code generators to strengthen their skills.

Correspondence:

Tanja Krunić

e-mail:

krunic@vtsns.edu.rs



This was our motivation for the research focused on using AI tools for code generation. The selected technology for which code is to be generated is MongoDB. The paper is organized as follows: Section 2 gives a brief description of MongoDB databases; using AI for code generation is presented in Section 3; the research of using an AI tool for generating MongoDB queries is briefly described in Section 4 together with a discussion of the results.

2. MONGODB DATABASES

MongoDB is a NoSQL database. This means that the storage data is not in a table-like format. Namely, data in MongoDB databases are stored in documents that consist of fields (key-value pairs) where values can be of different types (such as string, number, array, object, documents, arrays of documents, etc.). This is enabled since documents have a flexible structure of a variant of JSON called BSON (Binary JSON). These fields are analogous to columns in SQL databases. Any document has its unique identifier. Documents are stored in collections analogous to tables in SQL databases. A MongoDB database can contain more collections [7].

MongoDB is open source and free to use for applications written in many popular languages like C, C++, C#, .Net, Go, Java, Node.js, PHP, Python, etc. The complete list of languages that have MongoDB driver support can be found in [8].

One of the main advantages of using MongoDB databases is that they can be used for storing a large amount of data. This is why many world-famous companies like Facebook, eBay, Adobe, Google, Toyota, Uber, Bosch, Expedia, and Forbes use them in their web applications, [9], [10]. Another advantage of using MongoDB is claimed to be the flexibility of data structure – namely, the structure can change over time, and different documents can have different fields, [11]. In other words, MongoDB is a schema-less database, [9]. Its high performance is also at the top of the list of advantages of MongoDB. Other advantages can be found in [9] - [11].

There are many MongoDB GUIs available, like MongoDB Compass, Studio 3T, Robo 3T, MongoDB Atlas, NoSQLBooster, MongoDB for VS Code, and TablePlus, [12]. For this paper, MongoDB Compass [13] is used as a MongoDB GUI. It is a visual environment for querying, aggregating, and analyzing data [14]. Compass provides a visual representation of collections, which helps a better understanding of the document structure and

makes it easier to write queries.

3. AI code generation

AI code generation presents a system that generates code upon a user's request. The user of this system is supposed to describe the purpose of the desired code and then the AI code generator tool responds with the desired code. The system uses AI and specific software tools to generate the code [15]. Gemini Code Assist [16], CodePal [17], AI Code Generator [18], and Code Convert [19] appear on the top of the Google search result page for the term 'AI code generator'. According to [18], [20], [21], some of the most popular AI tools for code generation are claimed to be: Mintlify, Debuild, Codiga, Tabnine, AskCodi, OpenAICodex, GitHub Copilot, Codeium, CodePal, and Mutable AI.

Source [22] claims that programming occupations are already starting to be impacted by AI and that this trend will very likely continue. AI code generation tools can create boilerplate code, assist in creating code snippets, save time by automating some repetitive coding tasks, write code documentation, and help in identifying and fixing code bugs. As using AI tools for code generation increases the productivity and efficiency of software developers, it is clear that these will reduce the number of available jobs. On the other hand, coding skills will lose value since a large amount of code can be generated using AI code-generation tools. This means that coders can lose jobs or their salaries can decrease significantly, [23]. However, AI will not reach the human problem-solving ability for a long time so code generation must be guided by programmers. Since using AI tools increases their efficiency, programmers should use them to remain competitive in the labor market, [24].

4. WRITING MONGODB QUERIES USING AI CODE GENERATOR

Using an AI code generator for programming seems very comfortable. But the truth is that we have to use it correctly. This means that anyone who uses an AI code generator for code generation should have a good knowledge of the programming language for which the code has been generated. This is the only way one can give precise instructions to the AI code generator to avoid errors. In other words, AI can guide a programmer, but should also be guided by the programmer. To illustrate this, research on using an AI code generator tool for writing MongoDB queries is conducted in this paper. The research methodology is described in the subsection below.



4.1. RESEARCH METHODOLOGY

MongoDB Compass framework is used to generate a company database containing client data. For this purpose, a sample Excel file from [25] is used and it is further modified for the research. Then it is converted to .csv format and imported to MongoDB Compass. Note that the database is intentionally not named company and its collection is not named clients. Instead, the database is called 'firma', while its collection is called 'klijenti'. This is done to confuse the AI tool since 'company' and 'clients' are commonly used names. The collection is created automatically by importing the aforementioned .csv file. The structure of a document within the observed collection is shown in Figure 1.

This research has aimed to use an AI tool to generate common requests to the database. For this purpose, AI code generator [18] is used. A set of 28 queries is generated using simple instructions from the programmer to the AI code generator.

In the case of incorrectly generated queries, the given instruction to AI is corrected to be more detailed. As one can see from Figure 2 Left, this tool gives us an option to enter the desired coding language (which is MongoDB in our case), and the description of the code we want to be generated. After clicking the 'Execute' button, the AI code generator returns the desired code (Figure 2 Right).

4.2. RESEARCH RESULTS AND DISCUSSIONS

The research results are given in Table 1. As can be seen, the main issue in creating correct MongoDB queries is to tell the AI code generator the name of the collection and to provide the document structure.

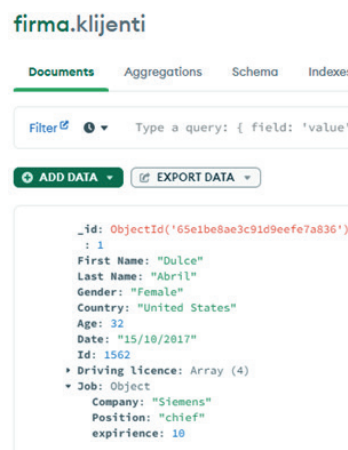


Figure 1. The structure of a document in the collection 'klijenti' of the MongoDB database.

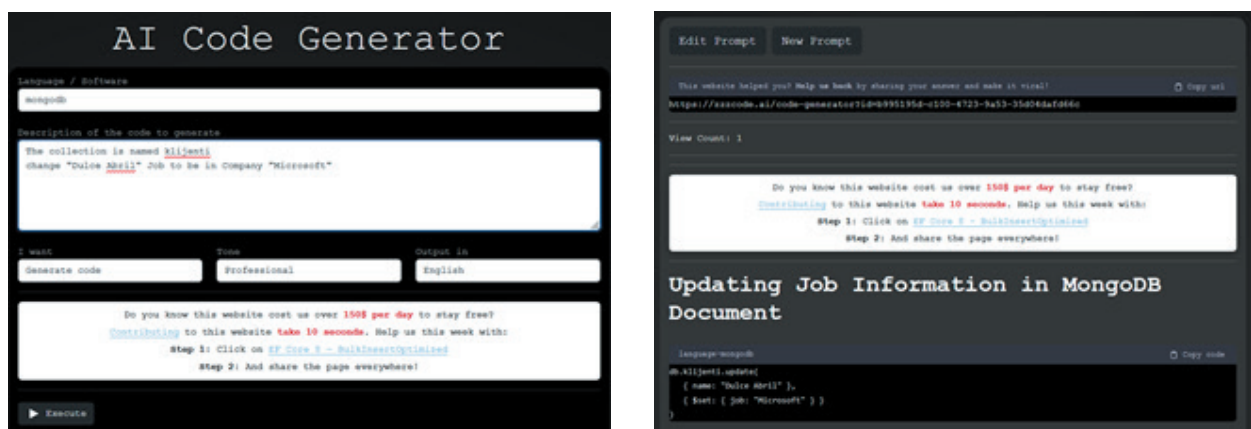


Figure 2. Left: A request to AI code generator, Right: The returned code.



Table 1. Requests to AI code generator and the obtained code.

No.	Request to the AI code generator	Obtained code	Accuracy	Comment
1.	Change "Dulce Abril" job to be in company "Microsoft"	<code>db.employees.updateOne({ name: "Dulce Abril" }, { \$set: { company: "Microsoft" } })</code>	-	The collection name should be given
2.	The collection name is kljenti change "Dulce Abril" job to be in company "Microsoft"	<code>db.klijenti.updateOne({ name: "Dulce Abril" }, { \$set: { company: "Microsoft" } })</code>	-	AI does not know that 'Job' is an object with the property 'Company'. Document structure should be given.
3.	Mandatory code snippet Change "Dulce Abril" Job to be in Company "Microsoft"	<code>db.klijenti.update({ "First Name": "Dulce", "Last Name": "Abril" }, { \$set: { "Job.Company": "Microsoft" } })</code>	+	
4.	Change the name "Sau" to "Betty"	<code>db.users.update({ name: "Sau" }, { \$set: { name: "Betty" } })</code>	-	Give the collection name and add a mandatory code snippet in every prompt. AI does not remember this data from previous prompts.
5.	Mandatory code snippet Change the name "Sau" to "Betty"	<code>db.klijenti.update({ "First Name": "Sau" }, { \$set: { "First Name": "Betty" } }, { multi: true })</code>	+	
6.	Mandatory code snippet If the country is France, add driving licenses A, B, and C	<code>db.klijenti.updateOne({ "Country": "France" }, { \$addToSet: { "Driving license": { \$each: ["A", "B", "C"] } } })</code>	±	This is correct, but we have to specify if we want to add A, B, and C driving licenses to all French people.
7.	Mandatory code snippet If the country is France, add driving licenses A, B, and C. Update all matches.	<code>db.klijenti.updateMany({ "Country": "France" }, { \$addToSet: { "Driving license": { \$each: ["A", "B", "C"] } } })</code>	+	
8.	Mandatory code snippet For all documents where the country is France, add a driving license D	<code>db.klijenti.updateMany({ "Country": "France" }, { \$addToSet: { "Driving license": "D" } })</code>	+	
9.	Mandatory code snippet Find people from France and give me names and driving licenses	<code>db.klijenti.find({ "Country": "France" }, { "First Name": 1, "Driving license": 1, "_id": 0 })</code>	+	
10.	Mandatory code snippet Find people from the United States and give me names and job positions	<code>db.klijenti.find({ "Country": "United States" }, { "First Name": 1, "Last Name": 1, "Job.Position": 1, "_id": 0 })</code>	+	
11.	Mandatory code snippet Find people from the United States and give me their names and job positions	<code>db.klijenti.find({ "Country": "United States" }, { "First Name": 1, "Last Name": 1, "Job.Position": 1, "_id": 0 })</code>	±	If you want only first or last name, you have to emphasize this.
12.	Mandatory code snippet Find people from the United States and give me their first name and job positions	<code>db.klijenti.find({ "Country": "United States" }, { "First Name": 1, "Job.Position": 1, "_id": 0 })</code>	+	
13.	Mandatory code snippet Delete one person with a C driving license	<code>db.klijenti.deleteOne({ "Driving license": "C" })</code>	+	
14.	Mandatory code snippet Delete one person working in Siemens	<code>db.klijenti.deleteOne({ "Job.Company": "Siemens" })</code>	+	
15.	Mandatory code snippet Insert a random document in this collection	<code>db.klijenti.insertOne({ "First Name": "John", "Last Name": "Doe", "Gender": "Male", "Country": "Canada", "Age": 30, "Date": "25/08/2021", "Id": 9876, "Driving license": ["A", "B"], "Job": { "Company": "Google", "Position": "Engineer", "experience": 5 } })</code>	+	



No.	Request to the AI code generator	Obtained code	Accuracy	Comment
16.	Mandatory code snippet Find all persons from France and sort them	db.klijenti.find({ "Country": "France" }).sort({ "First Name": 1 })	±	Specify the sort criteria. The first data is the default for sorting. A-Z sorting is used.
17.	Modified mandatory code snippet Find all persons from France and sort them	db.klijenti.find({ "Country": "France" }).sort({ "First Name": 1 })	+	
18.	Mandatory code snippet Find all persons from France, sort them, and count how many there are	db.klijenti.find({ "Country": "France" }).sort({ "First Name": 1 }).count()	+	
19.	Mandatory code snippet Find all persons from the United States, sort them by the company, and count how many there are	db.klijenti.find({ "Country": "United States" }).sort({ "Job.Company": 1 }).count()	+	
20.	Mandatory code snippet Delete for all people driving license A	db.klijenti.deleteMany({ "Driving licence": "A" })	-	This deletes all clients who have driving license A from the collection
21.	Mandatory code snippet Delete category A from the list of driving licenses	db.klijenti.update({}, { \$pull: { "Driving licence": "A" }, { multi: true } })	+	
22.	Mandatory code snippet For all people from Great Britain add children (number is three, names are Betty, Jill and Jill, gender is male, female and female, they are 5,7and 9 years old.	db.klijenti.updateMany({ "Country": "Great Britain" }, { \$set: { "Children": [{ "Name": "Betty", "Gender": "Male", "Age": 5 }, { "Name": "Jill", "Gender": "Female", "Age": 7 }, { "Name": "Bill", "Gender": "Female", "Age": 9 }] } })	+	Correct, even if not precisely prompted.
23.	Mandatory code snippet For all people from Great Britain add an object 'children' with properties "number", "names", "gender" and "age". There are three children. Their names are Betty, Jill, and Jill, Their gender is male, female, and female, they are 5 and 9 years old.	db.klijenti.updateMany({ "Country": "Great Britain" }, { \$set: { "children": [{ "number": 1, "names": "Betty", "gender": "male", "age": 5 }, { "number": 2, "names": "Jill", "gender": "female", "age": 7 }, { "number": 3, "names": "Jill", "gender": "female", "age": 9 }] } })	+	
24.	Mandatory code snippet Delete Jill from the list of children.	db.klijenti.update({ }, { \$pull: { "Children": "Jill" }, { multi: true } })	-	Children is a list of objects, not property.
25.	Mandatory code snippet Delete Jill from the object children.	db.klijenti.update({ }, { \$unset: { "children.Jill": "" }, { multi: true } })	-	
26.	Mandatory code snippet For all people from Great Britain from the array of objects children delete the one with the name Jill.	db.klijenti.update({ "Country": "Great Britain" }, { \$pull: { "children": { "name": "Jill" } }, { multi: true } })	-	
27.	Mandatory code snippet For all people from Great Britain from the array of objects children delete the one with the properties {Name: 'Jill', Gender: 'Female', Age: 7}	db.klijenti.update({ Country: "Great Britain" }, { \$pull: { children: { Name: "Jill", Gender: "Female", Age: 7 } }, { multi: true } })	-	This would be correct if the mandatory code was updated (The object should be "Children" instead of children)
28.	Updated Mandatory code snippet For all people from Great Britain from the array of objects children delete the one with the properties {Name: 'Jill', Gender: 'Female', Age: 7}	db.klijenti.update({ "Country": "Great Britain" }, { \$pull: { "Children": { "Name": "Jill", "Gender": "Female", "Age": 7 } }, { multi: true } })	+	



In Table 1, (see rows no. 1-5), we call the document structure 'Mandatory code snippet'. The mandatory code snippet for our example is given in Listing 1. As can be seen, the collection name is given in the comment of this snippet. The mandatory code snippet has to be provided in every prompt since AI does not remember the document structure from previous prompts.

Another important issue is to emphasize if we want to update one or all documents that match a filter. An example of this issue is given in row no. 6 in Table 1. In this example, the prompt to the AI code generator is to add driving licenses of A, B, and C categories if the country of a client is France. But it is not emphasized if we want to update only one or all matches. AI supposes that only one match is to be updated.

When using the sort function, we have to specify the sorting criteria. Otherwise, AI does A-Z sorting, as can be seen in the example given in row no. 16. One should also tell AI according to which field the results should be sorted. In our example, AI decides to sort the results according to the first field in the document, which is the first name. But when we switch places of the fields 'Gender' and 'First Name' (using a modified mandatory code snippet given in Listing 2), AI sorts the result

according to the first name data again; see row no. 17 in Table 1. In the example given in row no. 19, the sorting criterion is given to be the company of the client and AI responds with the correct code.

One should be very careful when giving prompts for deleting data. In the example given in row no. 20 of Table 1, when prompting 'Delete for all people driving license A', the AI code generator returns a query that deletes all documents in which the driving license of the category 'A' occurs in the list of driving licenses. If the user of the AI code generator does not realize this and tries the code directly on the database, it will cause an unwanted loss of data. To be more precise, one should give a prompt like 'Delete category A from the list of driving licenses' to the AI code generator. In this way, AI understands that 'A' is a category of driving licenses that should be deleted from the list of driving licenses in all documents. This indicates that we have to tell the AI code generator if a field value is an array, object, document, etc. Another example of this is given in row no. 24, where AI supposes the data is a property (and generates an incorrect query), although it is a list of objects, which we should have told AI.

```
//The collection name is 'klijenti'
//This is the structure of the document
{ "_id": { "$oid": "65e1be8ae3c91d9eefe7a839" },
  "First Name": "Kathleen", "Last Name": "Hanner", "Gender": "Female", "Country":
"United States", "Age": 125, "Date": "15/10/2017", "Id": 3549,
"Driving license": [ "A", "B", "C", "D"],
"Job": { "Company": "Siemens", "Position": "chief", "experience": 10 }}
```

Listing 1. Mandatory code snippet.

```
//The collection name is 'klijenti'
//This is the structure of the document
{"_id": { "$oid": "65e1be8ae3c91d9eefe7a839" },
  "Gender": "Female", "First Name": "Kathleen", "Last Name": "Hanner", "Country":
"United States", "Age": 125, "Date": "15/10/2017", "Id": 3549,
"Driving license": [ "A", "B", "C", "D"],
"Job": {"Company": "Siemens", "Position": "chief", "experience": 10}}
```

Listing 2. Modified mandatory code snippet.

```
//The collection name is 'klijenti'
//This is the structure of the document
{"_id": { "$oid": "65e1be8ae3c91d9eefe7a837"}, "First Name": "Mara", "Last Name":
"Hashimoto", "Gender": "Female", "Country": "Great Britain", "Age": 125, "Date":
"16/08/2016", "Id": 1582, "Driving license": [ "B", "C", "D"], "Children": [ { "Name":
"Betty", "Gender": "Male", "Age": 5 }, { "Name": "Bill", "Gender": "Female", "Age": 9}]}
```

Listing 3. Updated mandatory code snippet.



Also, examples no. 24-28 in Table 1 show that if we update the document structure by adding fields, then we also have to update our mandatory code snippet, because AI has to know the new document structure to create correct queries. The updated mandatory code snippet for our examples is given in Listing 3.

5. CONCLUSION

The occurrence of AI code-generating tools is making a revolution in coding. Coders can be faster, and more efficient and coding can become more comfortable with these tools. Using AI code generation tools will be mandatory for people who want to keep their jobs since using these tools is a significant advantage. On the other hand, any programmer who wants to use an AI code generator tool efficiently has to be familiar with the programming language used as the output of the request. Thus, he can describe the desired output in detail to get the correct code. After getting the code, one has to be able to distinguish if the code will give the desired result or not.

In the case of generating queries to MongoDB databases, the main issue is to inform the AI code generator tool about the structure of the document and the collection names. After changing the structure, AI has to be provided with the new structure of documents. It is also very important to emphasize to AI certain details of the desired result, for instance, if we want only one match with the filter criterion or all matches, and specify the desired sorting criterion. However, special attention has to be paid to delete prompts, since incorrectly generated delete prompts may harm our database.

6. REFERENCES

- [1] N. Laskowsky and L. Tucci, "What is artificial intelligence (AI)? Everything you need to know," [Online]. Available: <https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence>. [Accessed 6 April 2024].
- [2] "What is the history of artificial intelligence (AI)?," [Online]. Available: <https://www.tableau.com/data-insights/ai/history>. [Accessed 9 April 2024].
- [3] "What Is Artificial Intelligence (AI)?," [Online]. Available: <https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp>. [Accessed 29 March 2024].
- [4] "What are the most important advances in AI?," [Online]. Available: <https://ai100.stanford.edu/gathering-strength-gathering-storms-one-hundred-year-study-artificial-intelligence-ai100-2021-1/sq2>. [Accessed 29 March 2024].
- [5] "Artificial intelligence and its impact on everyday life," [Online]. Available: <https://online.york.ac.uk/artificial-intelligence-and-its-impact-on-everyday-life/>. [Accessed 25 March 2024].
- [6] B. Lutkevich, "Will AI replace jobs? 9 job types that might be affected," 3 November 2023. [Online]. Available: <https://www.techtarget.com/whatis/feature/Will-AI-replace-jobs-9-job-types-that-might-be-affected>. [Accessed 2023 December 2023].
- [7] A. S. Gillis and B. Botelho, "MongoDB," [Online]. Available: <https://www.techtarget.com/searchdata-management/definition/MongoDB>. [Accessed 12 February 2024].
- [8] "Start Developing with MongoDB," [Online]. Available: <https://www.mongodb.com/docs/drivers/>. [Accessed 17 October 2023].
- [9] "What is MongoDB – Working and Features," June 2021. [Online]. Available: <https://www.geeksforgeeks.org/what-is-mongodb-working-and-features/>. [Accessed 15 January 2024].
- [10] F.-J. Leick, "What is MongoDB and When is it a Good Business Choice?," [Online]. Available: <https://www.devoteam.com/expert-view/what-is-mongodb-and-when-is-it-a-good-business-choice/>. [Accessed 13 March 2024].
- [11] "What Is MongoDB?," [Online]. Available: <https://www.mongodb.com/company/what-is-mongodb>. [Accessed 11 April 2024].
- [12] A. Beazell, "The best MongoDB GUIs in 2023," 9 June 2020. [Online]. Available: <https://retool.com/blog/the-best-mongodb-guis>. [Accessed 28 December 2023].
- [13] "Compass. The GUI for MongoDB," [Online]. Available: <https://www.mongodb.com/products/tools/compass>. [Accessed 7 August 2023].



- [14] "What is MongoDB Compass?," [Online]. Available: <https://www.mongodb.com/docs/compass/current/>. [Accessed 4 April 2024].
- [15] "AI code generation," [Online]. Available: <https://www.sonarsource.com/learn/ai-code-generation/>. [Accessed 4 January 2024].
- [16] "Gemini Code Assist," [Online]. Available: <https://cloud.google.com/products/gemini/code-assist?hl=en#ai-assisted-application-development>. [Accessed 2 April 2024].
- [17] "CodePal," [Online]. Available: <https://codepal.ai/>. [Accessed 2 April 2024].
- [18] "AI Code Generator," [Online]. Available: <https://zzzcode.ai/code-generator>. [Accessed 7 November 2023].
- [19] "CodeConvert," [Online]. Available: <https://www.codeconvert.ai/free-code-generator>. [Accessed 9 April 2024].
- [20] A. McFarland, "10 Best AI Code Generators (April 2024)," 1 April 2024. [Online]. Available: <https://www.unite.ai/best-ai-code-generators/>. [Accessed 9 April 2024].
- [21] "24 Best AI Code Generators of 2024: Free and Paid," 24 September 2023. [Online]. Available: <https://blog.fabrichq.ai/latest-ai-code-generators-a-comprehensive-list-b7ce1a461fac>. [Accessed 2 April 2024].
- [22] R. Sahu, "Will Programmers Lose Their Jobs Because of AI?," 17 August 2023. [Online]. Available: <https://www.c-sharpcorner.com/article/will-programmers-lose-their-jobs-because-of-ai/>. [Accessed 12 March 2024].
- [23] S. Gupta, "What happens to software jobs with the rise of Generative AI?," 1 May 2023. [Online]. Available: <https://www.linkedin.com/pulse/what-happens-software-jobs-rise-generative-ai-sachin-gupta>. [Accessed 11 March 2024].
- [24] N. Sharma, "12 AI Tools That Can Generate Code To Help Programmers," 1 April 2024. [Online]. Available: <https://www.analyticsvidhya.com/blog/2023/08/ai-code-generator/>. [Accessed 9 April 2024].
- [25] "Sample .xls and xlsx download," [Online]. Available: <https://file-examples.com/index.php/sample-documents-download/sample-xls-download/>. [Accessed 2 December 2023].