

# Model-Based Testing of Web Application: An SLR

Muhammad Shoaib Farooq<sup>1</sup>, Ayesha Areej<sup>1</sup>

<sup>1</sup>Department of Department of Computer Science, University of Management and Technology, Lahore, Punjab, Pakistan

\*Corresponding author: [Shoaib.farooq@umt.edu.pk](mailto:Shoaib.farooq@umt.edu.pk)

## ABSTRACT

Website applications surpass system-based applications because they are nature friendly and installation free. Additionally, model-based testing is more beneficial under website applications as it saves time and money. Moreover, it detects fault lines at a higher speed. With the commencement of web application testing and model-based testing, various tools and models have been introduced in the domain of web application. In this milieu, systematical identification got significant importance to analyze MBT. To analyze the MBT, articles published between the years 2000 to 2022 are collected. From the total of 6115 papers are analyzed and 43 papers are selected. The presented SLR of MBT for the web application presents a comprehensive review of all the models and tools proposed in the literature. Furthermore, the taxonomy is defined about the models and steps followed in MBT. Lastly, the challenges and gaps are addressed. Hence, it is also helpful for researchers to avert the obstacles in the field of MBT for website applications.

## KEYWORD:

Model, model-based, model-driven, testing, tool, model-based testing, SLR, website, web application.

## JOURNAL INFO

HISTORY: Received: October 11, 2021

Accepted: December 15, 2021

Published: December 31, 2021

## 1. INTRODUCTION

There has been exceptional growth and development in the use of web applications in our day to day life from the day the www(world wide web) was created (Li et al. 2014)[1]. The web world is one of the fundamental aspects and has been perceived as an influential origin for conveying programming services on the web (Dogan et al. 2014)[2]. Web applications are very useful almost for everyone likewise education or schooling, business, government, public sector, other industries and it matters in our daily social life (Xiaolin et al. 2014) [3]. Rapid expansion draws the greatest possible level of advantages with the least effort computerized testing for broadening the software quality is automatic + action (automation) testing. (Xu et al. 2012)[4]. Applications are two of two types dynamics user and static user[5].

Traditional testing of application contributes same objectives of testing, mostly theories and methods changed and fitting new approaches for testing[6]. Traditional testing techniques are not capable of websites; therefore, they miss other characteristics like multi-tier architecture a hyperlinked structure. On testing websites, defined work is completed to assure that the program's application matches its specification[7].

MBT is a technique to fit the challenges required in testing software. In MBT, behavior is tested by creating a model[8]. MBT is a technique to generate a test case model from a requirement[9]. According to Dalal, MBT depends on these fundamentals. One is the criteria of generating test, the second is a description of software behavior, and the last is tools that generate infrastructure. Some other important aspects are levels of testing, which are automation, and non-

automated steps complexity[9]. Test engineers create models and define structural characteristics of SUT. Testers create models in formal languages such as UML. For structural aspects, class diagrams, and behavioral aspects, state charts and activity diagrams can be used[10].

With the commencement of web application testing and model-based testing, various tools and models have been introduced in the domain of web application. In this milieu, systematical identification got significant importance to analyze MBT. To analyze the MBT for website applications 43 papers are selected out of 6115 papers. This paper presents an SLR to provide an overview of models and tools. Furthermore, the challenges and gaps are identified and addressed. At last the detailed taxonomy is described. Furthermore, it is also helpful for researchers to avert the obstacles in the field of MBT for website application.

The research paper is structured as the related work is discussed of the already published studies related to the proposed article in section 2. Research methodology is provided in Section 3. Section 4 has discussed the data analysis part. Section 5 detailed the addressed questions area, and finally, section 6 discusses the conclusion of the paper.

## II. RELATED WORK

To the best of my knowledge, only five SLR in the literature discussed the MBT or web application testing but not MBT for the web application.

Serdar Dogan [2] proposed different models and identified fault models and implementation on the web application. They proposed a taxonomy of different fault models about browser conflict and synchronization. But



focus only on functional testing of a web application, and another limitation is it define model base testing on a very confined scale[2].

A survey by Mahesh Shirole and Rajeev Kumar [11] proposed various model-based testing approaches that are restricted into theoretic groups, formal languages, UML. But web application testing was discussed on a very narrow scale and not discussed the challenges and gaps in MBT of the web application.

In the survey By Mohamed Mussa [12], the analogy is made following the parameters such as UML, the antiquity of system design, and generation of test cases but oversight of the web applications.

Arilo et al. [13] define different MBT approaches for various specific domains such as web applications and COTS. But only focuses on the approaches, not on tools and models.

Hasan Javed and Nasir Mehmood Minhas [3] presented an SLR in which MBT approaches were identified, analyzed areas in which approaches were addressed, and defined tools supporting these techniques. But not discuss the challenges and gaps of MBT for web application testing.

The above analysis reveals that many researchers presented SLR and surveys on MBT approaches, techniques, and tools but did not discuss quality aspects, types of models and tools, challenges and gaps, and taxonomy as a major focus of their study. Proposed SLR is novel because it helps researchers give an overview of the models, tools, challenges, and gaps and a detailed taxonomy of MBT for the web application.

### III. RESEARCH METHODOLOGY

Systematic literature review produces a complete strategy to facilitate gathering, analysis, and identify gaps and resolution of primary articles in all current studies in the being studied field. The procedure guidelines for SLR schemed by B. Kitchenham [14] were followed to collect data impartially and demonstrate the results analyzed and cited in this study.

For this SLR, the review process being adopted is represented in Figure 1. It is a proper and thoughtful review procedure consist of six levels: 1)Define objectives of review 2)define questions of review 3)formulate the strategy of search 4)Papers screening and selection 5)using keywords analysis of papers 6)Extract and structure of information[14].

#### A. RESEARCH OBJECTIVES (RO)

The superior goal of this review is given below:

RO1: Explore and analyze quality aspects of testing web applications.

RO2: Investigate and synthesize test models and tools.

RO3: Listing of publication channels for the article published in MBT of the web application.

RO4: Construct a taxonomy of MBT and implementation of the testing process.

RO5: Challenges and gaps of existing models with model-based testing.

#### B. RESEARCH QUESTIONS (RQ)

To conduct SLR successfully, and at the beginning, the multiple review questions are defined. In addition, a complete search plan required for identifying the most important articles has been well-established in review. Review questions answered with the main motivation are listed in the table given below. The questions are answered according to the distinct procedure adopted in [14].

TABLE I. RQ and major motivations

	Research Question	Major Motivation
RQ1	What quality aspects of testing web applications have been considered important given in literature?	To discuss the quality aspects of MBT for web applications.
RQ2	What type of test models and tools has been discussed in the literature?	To find the index of test models and tools has been proposed.
RQ3	What are the challenges and gaps of existing models with model-based testing, found in literature?	To identify the challenges that MBT facing in existing models of selected domains.
RQ4	Which publications channels are primarily intended for the article published in MBT of web application in literature?	To discover the basic conferences and journals sources from where the dataset and literature of the chosen field of study can be published.

#### C. SEARCH SCHEME

A most critical step of the literature review(SLR) is the construction of a search scheme to determine and assemble likely outstanding papers in the selected field. That procedure includes defining the search string, literature resources, and a sorting system to find the most related articles in the cluster. Plenty of aspects of composing papers was empirically and qualitatively evaluated and accessed to represent the different aspects comparable with the research. In this phase, the scheme is decided to search for the relevant topic in the area of research.

##### 1) SEARCH STRING

Powerful research has been done by creating a string to collect the search available literature using various digital research collections in this field. To ensure the validity of



string about the validity of its outcome, key concepts were analyzed under consideration of the review questions to determine the appropriate keywords used in the selected field of research. The selected digital libraries are Springer link, google scholar, Elsevier, and IEEE. This step is to classify the logical scheme and search keywords to discover technological documentation in digital libraries. To identify maximum relevant articles, the complete keywords and their alternatives are required to create a string stated in Table II. The logical operators “AND” and “OR” are used to associate the selected and alternative keywords with formulating a string to search. Logical operator “OR” permit more options to explore. On the other hand, “AND” joins the terms to characterize the search choices to get the results.

**TABLE II. Terms and keywords used in search**

Database	String
Google scholar, IEEE, Elsevier, ACM, and Springer.	String (“Model-driven testing” OR “Model-based testing” OR “MBT” AND (“web application” OR “website application”))

**2) LITERATURE RESOURCES**

The field-specific journals are selected to manage the search of literature from online repositories for collection. Complete detail of the repository selected search string results that are applied are represented in Table III.

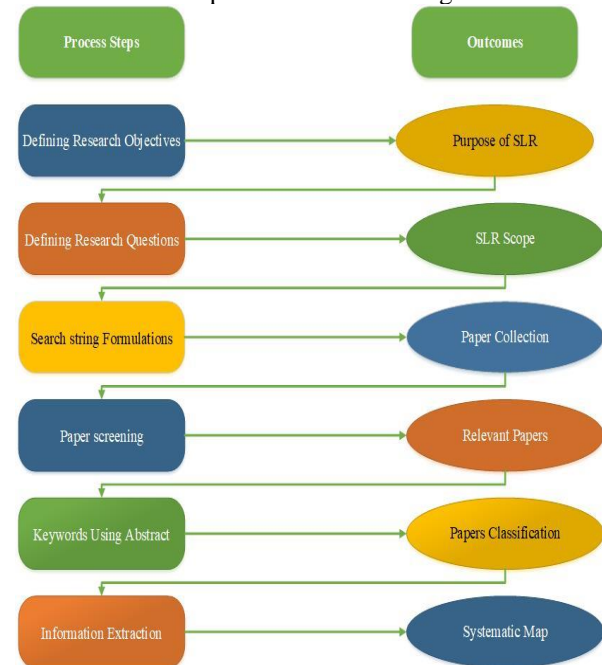
**TABLE III. Search strings of all selected publisher**

Repository	Search Strings
Springer Link	((“MODEL-BASED”) OR (“MODEL-DRIVEN”) AND (“TESTING” OR “TESTING TECHNIQUES”) AND (“WEB APPLICATION”) AND NOT (“REQUIREMENT ENGINEERING”))
Science Direct	((“MODEL-BASED”) OR (“MODEL-DRIVEN”) AND (“TESTING”) AND (“WEB APPLICATION”) NOT (“REQUIREMENT ENGINEERING”))
IEEE Xplore	(((((“ALL METADATA”:“MODEL BASED TESTING”) OR “ALL METADATA”: “MODEL DRIVEN TESTING”) AND “ALL METADATA”: WEB APPLICATION) NOT “ALL METADATA”: REQUIREMENT ENGINEERING)
ACM Digital Library	(((((“MODELS” OR “MODEL”) AND BASED) OR (“MODEL” AND “DRIVEN”) OR (“MODEL DRIVEN”)) AND (“WEB” OR “WEBSITE”) AND (“APP” OR “APPLICATION”)) OR (WEB APPLICATION))) NOT (“RE-ENGINEERING” OR (“REQUIREMENT AND “ENGINEERING”))

**D. SELECTION OF RELEVANT PAPERS**

During the search, all the papers are not absolutely according to review questions; so, they should be selected according to the relevance. For selection, the publication period to search is selected from 2000 to 2022.

By primarily searching various articles of research produced, not all are according to the review questions and duplicated. Search articles need screening to collect relevant papers. We used the process defined in [15] to screen articles. In step one, screening is done based on primary search and papers selected by their title and duplication removed. After that step, many irrelevant papers are available that are not according to the chosen domain. In the next selection, process abstracts are reviewed of every article selected in phase one of screening.



**FIGURE 1. SLR process**

**E. SCREENING BASED ON ABSTRACT KEYWORDING**

Furthermore, the screening of the papers has been prescribed by using the keyword method, which is deployed on a two-step abstract. To recognize the basic idea of the paper, the abstract was discussed initially and selected papers. From various articles, keywords are identified, combined, and then developed an amazing comprehension about the contribution in the research about chosen field. We utilized a process to find relevant papers by keyword utilizing the abstract. Keywording was completed in two stages. In the first step, we analyzed the abstract and found the topics and keywords that reflected the study's contribution. A greater degree of comprehension is formed in the second phase based on these keywords. We used keywords to build and cluster the categories for the review mapping.



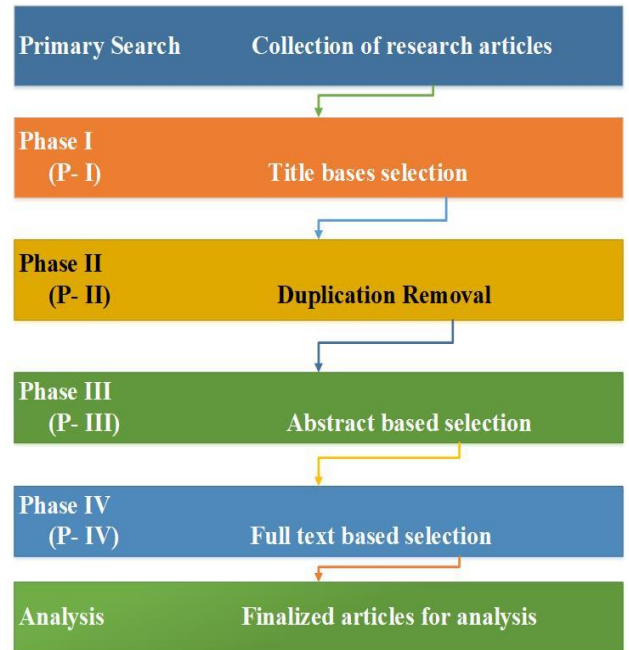
**F. QUALITY ASSESSMENT CRITERIA**

Quality assessment (QA) is often performed in an SLR to approach the quality of chosen papers. A questionnaire is created for this paper to approach the quality of chosen articles. (a) The research contributes to the model-based testing of a web application. "Yes (+1)" and "No (0)" were the two available responses to this study question. (b) Using MBT, the article provides a fair clarification in the field of testing. "Yes (+1)," "partially (0.5)," and "no (0)" were the possible responses to this research question. (c) Is the study published from a recognized source of publication? The JCR journal list and CORE conferences must be considered to answer this question. A systematic review's most crucial role is to assess the quality of the articles that have been chosen. Because the designs of these investigations differ, the quality of these studies was assessed using the mean of the sequential evaluation approach.

**TABLE IV Questionnaire to assess the quality**

Sr.	Assessment Questions	Expected Answers	Score
<b>Internal Scoring</b>			
1	Abstract was well defined?	a. Yes b. Intermediate c. No	a. 1.5 b. 1 c. 0
2	Review of the literature was explained in detail?	a. Yes b. Intermediate c. No	a. 1.5 b. 1 c. 0
3	Data collection was defined clearly?	a. Yes b. Intermediate c. No	a. 1.5 b. 1 c. 0
4	The description/collection of features was clearly defined?	a. Yes b. Intermediate c. No	a. 1.5 b. 1 c. 0
5	The methodology part was defined clearly?	a. Yes b. Intermediate c. No	a. 1.5 b. 1 c. 0
6	The conclusion was reasonable and completely result-based?	a. Yes b. Intermediate c. No	a. 1.5 b. 1 c. 0
<b>External scoring (based on publication source)</b>			
9	The research is published in the CORE ranking conference, symposium, and proceedings.	a. CORE A b. CORE B c. CORE C d. No CORE ranking	a. 2 b. 1.5 c. 1 d. 0
10	The research published	a. Q1	a. 2

in "Journal citation report" listed ranked journal	b. Q2 c. Q3/Q4 d. No JCR ranking	b. 1.5 c. 1 d. 0
--	--	------------------------



**FIGURE 2. Selection procedure steps**

**IV. DATA ANALYSIS**

For data extraction that is necessary to answer the review questions, reading of all selected papers in primary search is compulsory. Articles that were selected and checked to answer the review questions adequately. Firstly, discuss the result of the primary search on the initial search string and then the definition of assessment score; after that, complete discussion to answer the review question.

**A. SEARCH RESULTS**

Model-based testing of web applications involves using online databases for collecting the dataset. The process of primary search collects a total of 6115 articles from different databases.

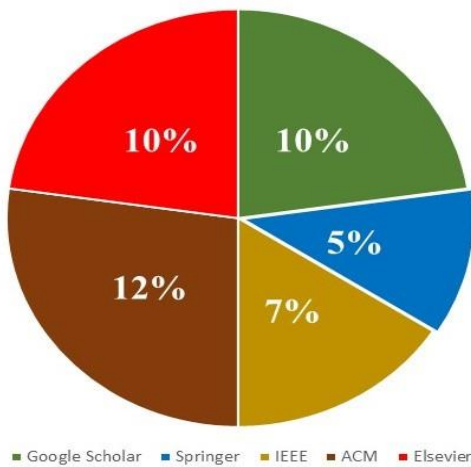
Databases	Primary search	P1	P2	P3	P4
Google scholar	4600	170	90	16	10
Springer	270	135	70	9	5
IEEE	500	240	123	13	7
ACM	465	250	150	15	12
Elsevier	280	151	67	7	10
Total	6115	946	500	60	43



Apply selection process on the data described in the above section. Results of selection by different phases are explained in Table V.

**TABLE V. Publisher Wise Selection Process**

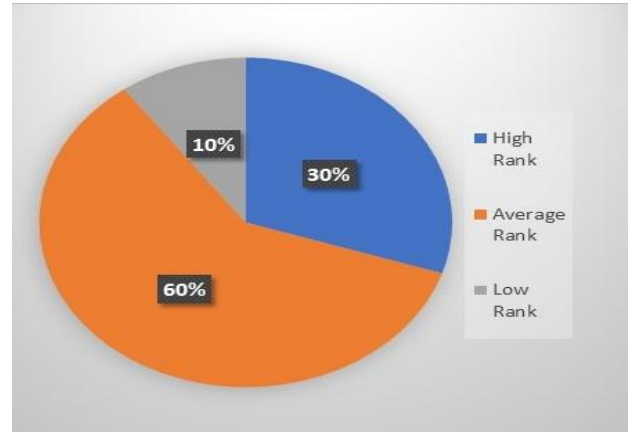
The primary selection is performed in primary search, and the 6115 articles are found. As a result, in (P-1), select articles on the base of title, and 946 articles are found. Now in (P-2), remove duplication and articles not relevant to the domain. The inclusion and exclusion-based search also applied in this phase, and 500 articles were found. In (P-3), abstract-based search is applied, and 60 papers are selected. The complete text-based screening was applied to 60 articles, and 43 were selected in phase 4. These articles are applicable and completed for including in research for data study and abstraction.



**FIGURE 3. Percentage of chosen studies database wise**

**B. QUALITY ASSESSMENT SCORE**

Scores are allocated to the selected studies according to the detailed scoring procedure calculated based on internal and external criteria given in Table VI. By internal and external criteria, which score is attained in Table VI defined as I-Score and E-Score. From all selected studies, 30% high score achieved high-ranking articles, 60% of articles achieved average ranking, and 10% ranked as low ranking as per the criteria.



**FIGURE 4. Score-based ranking percentage**

**TABLE VI. Quality assessment score and classification**

R e f .	P. Type	Domain	I- Scor e	E- Scor e	Tot al Sco re
[1 5]	Journal	MDE	6.5	2	8.5
[1 6]	Journal	MDE	7.5	2	9.5
[1 7]	Journal	PIM	6.0	1	7.0
[3 ]	Journal	MDD	8.0	2	10. 2
[1 4]	Journal	MDE	6.0	1.5	7.5
[1 8]	Journal	MDD	4.5	2	6.5
[2 ]	Confere nce	MDE	5.5	2	7.5
[1 9]	Journal	PIM	6.5	2	8.5
[1 3]	Journal	PIM	7.5	2	9.5
[2 0]	Journal	PIM	7.0	2	9
[2 2]	Journal	MDE	5.5	2	7.5
[2 2]	Journal	MDD	6.5	1.5	8
[2 3]	Confere nce	PIM	6.5	2	8.5
[1 ]	Confere nce	MDD	7.0	2	9
[2 4]	Journal	MDD	6.5	1.5	8
[4 ]	Confere nce	PIM	5.5	2	7.5
[1 1]	Journal	MDE	7.0	2	9



[2 5]	Confere nce	MDE	6.5	1.5	8
----------	----------------	-----	-----	-----	---

## V. DISCUSSIONS

We elaborate on our discoveries against our research questions in this part.

### 1) QUESTION 1 ASSESSMENT. WHAT QUALITY ASPECTS OF TESTING WEB APPLICATION HAS BEEN CONSIDERED IMPORTANT GIVEN IN LITERATURE?

MBT appears to be a good method for creating test scenarios from models of the SUT. To depict the conduct of a framework, these models are utilized. The information obtained from the requirement or stated document is used to construct the model. Many authors use MBT to test web apps. Recently published[2] systematic literature review on online application testing that compare various models, such as model types and faulty models, empirical investigations, and web application implementation. They use models of error and taxonomy as results in this survey. According to this, SLR is primarily used in literature to measure effort/test time. They also examine new trends in web application testing and provide practitioners and researchers with potential study areas.

This SLR's limitation is that it solely addresses the functional testing of web apps. Another downside is that it only supports MBT for small-scale web applications. Another survey on UML behavior models was presented by Mahesh Shirol and Rajeev Kumar [11]. Graph-based, formalized specification languages (OCL), and UML specifications are the three types of MBT techniques in this investigation (sequence diagrams, state diagrams, activity diagrams). Web application testing, but only to a limited extent, will be covered in this survey by the author. ACDNGHT surveyed the characterization of attributes for the MBT such as (behavioral model, the scale of difficulty, MBT approach indicator, the indication of support tools, and so on). Test based on models. They suggest future directions of investigation, such as model surveys, which is why we conducted this study[19].

Vahid Garousi et al. [26]. An intensive mapping study showed that a check for net applications was given. During this study, they checked out approaches, tools, and levels of tests for web applications. They organized the analysis findings into 5 categories appreciated forms of papers that supported the tributary side (based on contribution); these parts are their research topics for her mapping study included the following paper types: research facet (solution proposals), adequately testing (coding or modelling coverage), test areas (test facet such as users or servers), and testing level. The study's limitation in the context is that it solely appearance at MBTesting on a tiny low scale. A survey of model-driven procedures was presented by Mohamed Mussa et al. [12].

During this survey, various parameters appreciate modelling language (UML), system style artefacts, action at law generating, testing target, and gear support are compared. Web applications don't seem to be enclosed in this study. Eddy physiologist et al. [27] an analysis of model-based testing from UML models were presented. Among the UML models used in their research were categorical drawings, for example, visualizations and state machines. Transition-based diagrams, decision-based diagrams, and data-oriented criteria are wont to opt for check subjects. This paper's weakness is that it solely appearance at some forms of activity models, ignoring others who appreciate FSM, EFSM, directed and undirected graphs, and algorithms that we cowl in our overview[1]. The benefits of model-based testing are that it increases detection of faults, advanced test case design can systematically approach requirement coverage, intangible benefits[28]. Different examinations show the significance of website application testing. According to the findings of a study[2], 70 percent of online apps have flaws. Cost-effective and efficient testing techniques are required due to the multi-lingual nature of web applications and cross-platform dynamics [25].

### 2) QUESTION 2 ASSESSMENT. WHAT TYPE OF TEST MODELS AND TOOLS HAS BEEN DISCUSSED IN THE LITERATURE?

In support of this research question, various models are presented in web applications for various purposes, such as Statecharts [29]. On undirected graphs, the greedy algorithm is applied, shortest path graphs[30] FSM, Algorithms are applied[31]. FSM that is extended [32]. Navigation-based models created by Directed Graphs [33]. An algorithm, Finite State Machines [34]. Finite State Machine that is parametrized, FSM that is extended [22]. Ontologies, FSM[35].

To improve the blind usability of web applications, the shortest path graph, and weighted directed graph were used to analyze the navigability of online shopping sites[36].

The algorithm is adept at assuming an extended FSM of the web application. Z- quotient approach used and limitations removed, by using page clustering inputs and outputs are generated. Point and click model interference tool obtained, but obligations are identified on the model and then completed on application[37].

The structural model was designed using methods of reengineering to demonstrate the statistical features of the web application. For testing the dynamic features of a website application and model graphs and several ontologies and tools for mapping are used, test cases to the filling of forms are developed automatically [38].

For model-based web applications, a straightforward technique uses a finite state machine. The solution is to reduce the number of transitions that use compulsion on inputs thus this compressing the FSM[25].



How to model the system's workload is such an important task, especially when old models of workload have a compulsion. For generating manufactured workload of website CbSAM introduced. Activity diagrams and finite state machines are used in this LTF of web application[21].

To analyze dynamic code automatic test generation algorithm uses the run time values and creates models of the semantics of string operations. To resolve multiple types of compulsions by identifying each variable individually algorithm presented, also Finite State Machines[20].

Principles of the approach based on mixed SUT modelling: define WA's behavioural aspects, including VTP for test generation algorithm and MBVT approach adorned. In modelling form state chart diagrams and UML class diagrams are represented[39].

The presentation layer of the website application testing transition machines are used in the creation model[40]. Labelled transition machine used to model MBT web application[8]. UML activity diagrams[41].

The following tools are used for MBT web application: Test modeller, test optimal, Tricentis Tosca, SpecExplorer, Graph walker, NModel, Tcases, ModBat, MoMuT, Crawljax[28].

AETG, JUMBL, Microsoft Spec Explorer are tools used in literature to model base testing of website applications [42]. Swing, Selenium is a tool to test web applications in MBT[43].

### 3) QUESTION 3 ASSESSMENT. WHAT ARE THE CHALLENGES AND GAPS OF EXISTING MODELS WITH MODEL-BASED TESTING, FOUND IN LITERATURE?

Complex web apps will be the norm in the future. Modifiability by self, observability level so low, information is asynchronous, behaviour is time, and load-dependent are projected to be among their properties. Going on testing methodologies cannot guarantee the quality of these applications' faults for each feature. For example, testers using present testing approaches manually define test cases, and the quantity of resources required makes this process difficult to complete[44].

Reading the criteria and determining the limitations was the most difficult part of the modelling process. Some limitations are not addressed in the requirements. We had to test the application to check how it worked. This is to be expected, given that many applications we design have incomplete requirements. It forces developers to make decisions about the missing cases[9].

For UML models, some studies have provided design patterns on finest practices for MBT models; we found an ordinary lack of information and resources. Experienced team members review on a regular basis and inspection was our technique for quality assurance of model-based testing models. Still, design patterns are

needed for this ambition to help the testers for MBT design to develop good quality MBT models[28].

The testing procedure suggested as the main purpose is to create model of the web application and from the model select the test cases. The web application is separated into divisions based on its functionality and structure. In the first stage of our approach to avoid complexity and produce a model of major items of the application. As a result, the produced sections can be verified using the application's analysis and documents implementation[15].

The fundamental disadvantage of MBVT is the same as that of MBT is normally: the time and effort required to construct models, test purposes, and adapters [45].

The worth of web applications can be integrated dynamically, which poses a significant difficulty for analysis and testing. Someday, we plan to resolve the issue of dynamic integration in the framework of this paradigm. State-based behaviour, numerous users, and consistency issues have not been addressed[40].

We plan to expand our web application fault analysis to encompass additional types of interactive software in the future, such as graphical user interfaces[46].

1) broadening the scope of the approach to include additional technical and logical vulnerability classes. We'll also look at 2) from one project to the next, how to make the various MBVT antiquity reusable. In the present situation, we will concentrate on apps about eCommerce, namely eCommerce applications built on the Magento platform[39].

The most tricky part is requirements reading and constraints determination in the construction of the model. Indeed, most of the constraints are not read in requirements reading then for reading them; the application has to be run to see what has been implemented. It analyzes behavioural and input models in combination to reach effectiveness and input domain coverage [9].

From the benefits and needs, model-based testing now faces several difficulties. First, the test case explosion problem can be caused by automatic generation, driven by coverage criteria. Another issue with the current MBT is the lack of support for non-functional testing. Few people look at non-functional testing like performance, load, and usability testing[10].

Gaps are identified for future researchers and aim for further enhancement in this research area[3].

### MODEL AND STEPS USED IN MODEL-BASED TESTING:

The presented taxonomy in Figure 5 shows the testing models for MBT of web applications and the steps for the testing process. In the first step, types of models are defined, then construct a model in specification criteria like subject, characteristics, and paradigm. The first dimension is the subject of the model. The input/output models of the SUT(system under test) show the valid input that can be sent to the SUT(system under test) and capture the



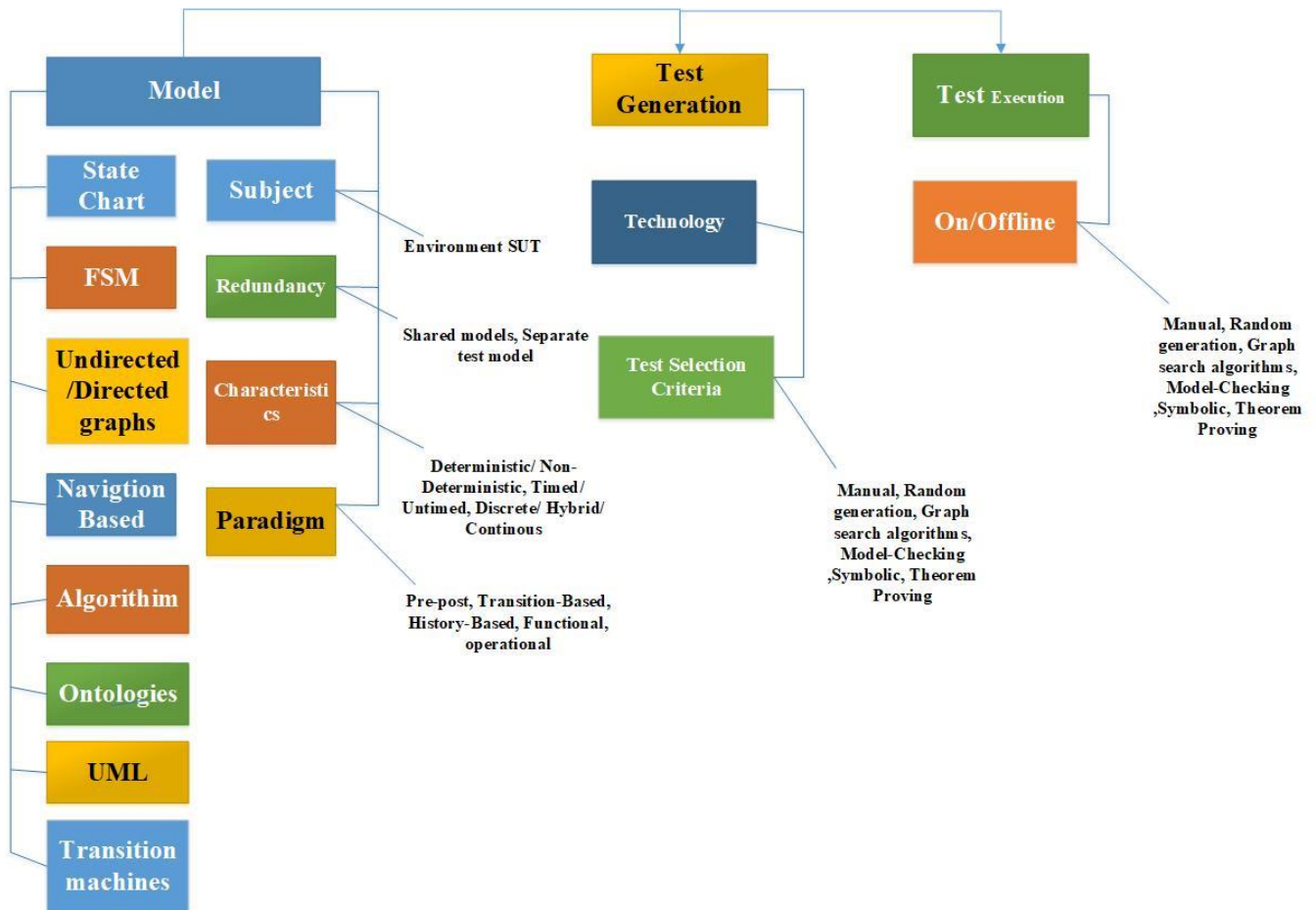
SUT(system under test) behaviour that is intended. In the test generation, step technology and criteria of test selection are defined because criteria of test selection and technology reflect generating tests. In the third step, test execution is reflected by the offline and online dimensions.

4) QUESTION 4 ASSESSMENT. WHICH PUBLICATION CHANNELS ARE PRIMARILY INTENDED FOR ARTICLE PUBLICATION IN MBT OF WEB APPLICATIONS IN LITERATURE?

The presented research question is written to obtain the beneficial publication source to collect the articles according to the domain. Finally, publication channels of good ranking articles have been found to support researchers for further study in the form of a list. Table XI summary of the primary intended study for this article in JCR and core ranking sources has been presented for the quality assessment of all the articles.

**TABLE XI.** Summary of publication channels and their ranking of selected studies

Publication Source	JCR / CORE Rank	Publication Type	References	No of Articles	Ratio (%)
Journal of system and software Information and Software Technology	Q1	Journal	[2][28]	2	5%
Information systems Egyptian Informatics Journal	Q2	Journal	[4][26][25][15][47][7][6]	7	19%
Software Testing, Verification and Validation	Q1	Journal	[1]	1	3%
ACM SIGSOFT Software Engineering (ESEM) Empirical Software Engineering and Measurement	Q1	Journal	[48]	1	3%
Journal of Software	A	Conference	[37][22][33][43][39]	5	14%
Software testing and analysis	A*	Conference	[11]	1	3%
Quality Software	A	Conference	[19]	1	3%
Software testing and analysis	Q1	Journal	[3]	1	3%
Computer-Human Interaction	A	Conference	[20]	1	5%
Open Systems	B	Conference	[12][21]	2	5%
Computer-Human Interaction	A	Conference	[20][46]	2	5%
Software testing, verification, and reliability	C	Conference	[36]	1	3%
Lecture Notes in Computer Science	B	Conference	[38][41]	2	5%
Software and Systems Modeling	Q2	Journal	[36]	1	3%
Journal of the Chinese Institute of Engineers	Q2	Journal	[42]	1	3%
International Conference on Software Engineering	Q3	Journal	[8]	1	3%
Advances in computers	Q2	Journal	[40]	1	3%
IEEE software	Q3	Journal	[5]	1	3%
Research challenges in information science	A*	Conference	[9]	1	3%
Australian Software Engineering	Q1	Journal	[10]	1	3%
	Q1	Journal	[49]	1	3%
	B	Conference	[44]	1	3%
	C	Conference	[35]	1	3%



**FIGURE 5. Taxonomy of Models and steps used in MBT**

**VI. CONCLUSION**

Web application testing is a very important aspect in developing an application for removing the complexity of web application services. Model-based testing is beneficial for such applications because it is highly good at fault detection, low cost, and less time-consuming. This paper aims to generate a list of models and tools and address the challenges and gaps. In this review, quality aspects are important, models used to perform MBT, and tools used in MBT of web applications. This proposed article also discusses the challenges and gaps and a detailed taxonomy. Publication channels and sources list created in the field of the selected study. All algorithms, models, UML, charts, and tools discussed to give researchers an overview for future research.

**REFERENCES**

[1] Y. F. Li, P. K. Das, and D. L. Dowe, "Two decades of Web application testing - A survey of recent advances," *Inf. Syst.*, vol. 43, pp. 20–54, 2014, doi: 10.1016/j.is.2014.02.001.  
 [2] S. Doğan, A. Betin-Can, and V. Garousi, "Web application testing: A systematic literature review," *J. Syst. Softw.*, vol. 91, no. 1, pp. 174–201, 2014, doi: 10.1016/j.jss.2014.01.010.  
 [3] H. Javed, N. Mehmood Minhas, A. Abbas, and F.

Muhammad Riaz, "Model Based Testing for Web Applications: A Literature Survey Presented," *J. Softw.*, vol. 11, no. 5, pp. 347–361, 2016, doi: 10.17706/jsw.11.4.347-361.  
 [4] L. Ran, C. Dyreson, A. Andrews, R. Bryce, and C. Mallery, "Building test cases and oracles to automate the testing of web database applications," *Inf. Softw. Technol.*, vol. 51, no. 2, pp. 460–477, 2009, doi: 10.1016/j.infsof.2008.05.016.  
 [5] X. Wang, B. Zhou, and W. Li, "Model-based load testing of web applications," *J. Chinese Inst. Eng. Trans. Chinese Inst. Eng. A*, vol. 36, no. 1, pp. 74–86, 2013, doi: 10.1080/02533839.2012.726028.  
 [6] G. A. Di Lucca and A. R. Fasolino, "Testing Web-based applications: The state of the art and future trends," *Inf. Softw. Technol.*, vol. 48, no. 12, pp. 1172–1186, 2006, doi: 10.1016/j.infsof.2006.06.006.  
 [7] N. Mansour and M. Hourri, "Testing web applications," *Inf. Softw. Technol.*, vol. 48, no. 1, pp. 31–42, 2006, doi: 10.1016/j.infsof.2005.02.007.  
 [8] J.-P. Katoen, "22 Labelled Transition Systems," pp. 615–616, 2005, doi: 10.1007/11498490\_29.  
 [9] A. Pretschner, "Model-based testing in practice," *Lect. Notes Comput. Sci.*, vol. 3582, pp. 537–541, 2005, doi: 10.1007/11526841\_37.  
 [10] M. Mlynarski, B. Güldali, S. Weißleder, and G. Engels, *Model-Based Testing: Achievements and Future Challenges*, vol. 86. Elsevier Inc., 2012.



- [11] M. Shirole and R. Kumar, "UML behavioral model based test case generation," *ACM SIGSOFT Softw. Eng. Notes*, vol. 38, no. 4, pp. 1–13, 2013, doi: 10.1145/2492248.2492274.
- [12] M. Mussa, S. Ouchani, W. Al Sammane, and A. Hamou-Lhadj, "A survey of model-driven testing techniques," *Proc. - Int. Conf. Qual. Softw.*, pp. 167–172, 2009, doi: 10.1109/QSIC.2009.30.
- [13] A. C. Dias Neto, R. Subramanyan, M. Vieira, and G. H. Travassos, "A survey on model-based testing approaches: A systematic review," *Proc. - 1st ACM Int. Work. Empir. Assess. Softw. Eng. Lang. Technol. WEASEL Tech 2007, Held with 22nd IEEE/ACM Int. Conf. Autom. Softw. Eng., ASE 2007*, pp. 31–36, 2007, doi: 10.1145/1353673.1353681.
- [14] B. Kitchenham, "Procedures for performing systematic reviews," *Keele Univ. Natl. ICT Aust.*, 2004, doi: 10.1.1.122.3308.
- [15] E. Habibi and S. H. Mirian-Hosseinabadi, "Event-driven web application testing based on model-based mutation testing," *Inf. Softw. Technol.*, vol. 67, pp. 159–179, 2015, doi: 10.1016/j.infsof.2015.07.003.
- [16] J. L. Herrero Agustin and P. C. Del Barco, "A model-driven approach to develop high performance web applications," *J. Syst. Softw.*, vol. 86, no. 12, pp. 3013–3023, 2013, doi: 10.1016/j.jss.2013.07.028.
- [17] S. Ouhbi, A. Idri, J. L. Fernández-Alemán, and A. Toval, "Requirements engineering education: a systematic mapping study," *Requir. Eng.*, 2015, doi: 10.1007/s00766-013-0192-5.
- [18] A. Fernandez, E. Insfran, and S. Abrahão, "Usability evaluation methods for the web: A systematic mapping study," in *Information and Software Technology*, 2011, vol. 53, no. 8, pp. 789–817, doi: 10.1016/j.infsof.2011.02.007.
- [19] A. C. Dias Neto and G. H. Travassos, "Surveying model based testing approaches characterization attributes," *ESEM'08 Proc. 2008 ACM-IEEE Int. Symp. Empir. Softw. Eng. Meas.*, pp. 324–326, 2008, doi: 10.1145/1414004.1414069.
- [20] G. Wassermann, D. Yu, A. Chander, D. Dhurjati, H. Inamura, and Z. Su, "Dynamic test input generation for web applications," *ISSTA'08 Proc. 2008 Int. Symp. Softw. Test. Anal. 2008*, pp. 249–259, 2008, doi: 10.1145/1390630.1390661.
- [21] J. Zhou, B. Zhou, and S. Li, "LTF: A model-based load testing framework for web applications," *Proc. - Int. Conf. Qual. Softw.*, pp. 154–163, 2014, doi: 10.1109/QSIC.2014.53.
- [22] K. Hossen, R. Groz, C. Oriat, and J. L. Richier, "Automatic generation of test drivers for model inference of web applications," *Proc. - IEEE 6th Int. Conf. Softw. Testing, Verif. Valid. Work. ICSTW 2013*, pp. 441–444, 2013, doi: 10.1109/ICSTW.2013.57.
- [23] M. N. Gabere and W. S. Noble, "Empirical comparison of web-based antimicrobial peptide prediction tools," *Bioinformatics*, vol. 33, no. 13, pp. 1921–1929, Jul. 2017, doi: 10.1093/bioinformatics/btx081.
- [24] X. Xu, H. Jin, S. Wu, L. Tang, and Y. Wang, "URMG: Enhanced CBMG-based method for automatically testing web applications in the cloud," *Tsinghua Sci. Technol.*, vol. 19, no. 1, pp. 65–75, 2014, doi: 10.1109/tst.2014.6733209.
- [25] A. A. Andrews, J. Offutt, C. Dyreson, C. J. Mallery, K. Jerath, and R. Alexander, "Scalability issues with using FSMWeb to test web applications," *Inf. Softw. Technol.*, vol. 52, no. 1, pp. 52–66, 2010, doi: 10.1016/j.infsof.2009.06.002.
- [26] V. Garousi, A. Mesbah, A. Betin-Can, and S. Mirshokraie, "A systematic mapping study of web application testing," *Inf. Softw. Technol.*, vol. 55, no. 8, pp. 1374–1396, 2013, doi: 10.1016/j.infsof.2013.02.006.
- [27] E. Bernard *et al.*, "Model-based testing from UML models," *Inform. 2006 - Inform. fur Menschen, Beitrage der 36. Jahrestagung der Gesellschaft fur Inform. e.V.*, vol. 2, no. May 2014, pp. 223–230, 2006.
- [28] V. Garousi, A. B. Keleş, Y. Balaman, Z. Ö. Güler, and A. Arcuri, "Model-based testing in practice: An experience report from the web applications domain," *J. Syst. Softw.*, vol. 180, p. 111032, 2021, doi: 10.1016/j.jss.2021.111032.
- [29] P. Bansal and S. Sabharwal, "A model based approach to test case generation for testing the navigation behavior of dynamic web applications," *2013 6th Int. Conf. Contemp. Comput. IC3 2013*, pp. 213–218, 2013, doi: 10.1109/IC3.2013.6612192.
- [30] A. Al-Herz and M. Ahmed, "Model-based web components testing: A prioritization approach," *Commun. Comput. Inf. Sci.*, vol. 181 CCIS, no. PART 3, pp. 25–40, 2011, doi: 10.1007/978-3-642-22203-0\_3.
- [31] M. E. Dincturk, G. V. Jourdan, G. V. Bochmann, and I. V. Onut, "A model-based approach for crawling Rich Internet Applications," *ACM Trans. Web*, vol. 8, no. 3, 2014, doi: 10.1145/2626371.
- [32] M. Shams, D. Krishnamurthy, and B. Far, "A model-based approach for testing the performance of Web applications," in *Proceedings of the Third International Workshop on Software Quality Assurance, SOQUA 2006*, 2006, pp. 54–61, doi: 10.1145/1188895.1188909.
- [33] S. Sprengle, L. Pollock, and L. Simko, "A study of usage-based navigation models and generated abstract test cases for web applications," *Proc. - 4th IEEE Int. Conf. Softw. Testing, Verif. Validation, ICST 2011*, pp. 230–239, 2011, doi: 10.1109/ICST.2011.34.
- [34] F. Belli, A. T. Endo, M. Linschulte, and A. Simao, "Model-based testing of web service compositions," *Proc. - 6th IEEE Int. Symp. Serv. Syst. Eng. SOSE 2011*, no. Sose, pp. 181–192, 2011, doi: 10.1109/SOSE.2011.6139107.
- [35] M. S. Jokhio, G. Dobbie, and J. Sun, "Towards specification based testing for semantic web services," *Proc. Aust. Softw. Eng. Conf. ASWEC*, pp. 54–63, 2009, doi: 10.1109/ASWEC.2009.38.
- [36] H. Takagi, S. Saito, K. Fukuda, and C. Asakawa, "Analysis of navigability of Web applications for improving blind usability," *ACM Trans. Comput. Interact.*, vol. 14, no. 3, 2007, doi: 10.1145/1279700.1279703.
- [37] K. Hossen, R. Groz, C. Oriat, and J. L. Richier, "Automatic model inference of web applications for security testing," *Proc. - IEEE 7th Int. Conf. Softw. Testing, Verif. Valid. Work. ICSTW 2014*, pp. 22–23, 2014, doi: 10.1109/ICSTW.2014.47.
- [38] H. Hajiabadi and M. Kahani, "An automated model based approach to test web application using ontology," *2011 IEEE Conf. Open Syst. ICOS 2011*, pp. 348–353, 2011, doi: 10.1109/ICOS.2011.6079282.
- [39] F. Lebeau, B. Legard, F. Peureux, and A. Vernotte, "Model-based vulnerability testing for web applications," *Proc. - IEEE 6th Int. Conf. Softw. Testing, Verif. Valid. Work. ICSTW 2013*, no. September 2015, pp. 445–452, 2013, doi: 10.1109/ICSTW.2013.58.
- [40] J. Offutt and Y. Wu, "Modeling presentation layers of web applications for testing," *Softw. Syst. Model.*, vol. 9, no. 2, pp. 257–280, 2010, doi: 10.1007/s10270-009-0125-4.
- [41] C. H. Huang and H. Y. Chen, "A tool to support automated testing for Web application scenario," *Conf. Proc. - IEEE Int. Conf. Syst. Man Cybern.*, vol. 3, pp. 2179–2184, 2006,



doi: 10.1109/ICSMC.2006.385184.

[42] M. Utting, A. Pretschner, and B. Legeard, "A taxonomy of model-based testing approaches," *Softw. Test. Verif. Reliab.*, vol. 22, no. 5, pp. 297–312, 2012, doi: 10.1002/stvr.456.

[43] T. Pajunen, T. Takala, and M. Katara, "Model-based testing with a general purpose keyword-driven test automation framework," *Proc. - 4th IEEE Int. Conf. Softw. Testing, Verif. Valid. Work. ICSTW 2011*, pp. 242–251, 2011, doi: 10.1109/ICSTW.2011.39.

[44] B. Marín, T. Vos, G. Giachetti, A. Baars, and P. Tonella, "Towards testing future Web applications," *Proc. - Int. Conf. Res. Challenges Inf. Sci.*, 2011, doi: 10.1109/RCIS.2011.6006859.

[45] A. Vernotte, "Research questions for model-based vulnerability testing of web applications," *Proc. - IEEE 6th Int. Conf. Softw. Testing, Verif. Validation, ICST 2013*, pp. 505–506, 2013, doi: 10.1109/ICST.2013.82.

[46] K. Dobolyi and W. Weimer, "Modeling Consumer-Perceived Web Application Fault Severities for Testing Categories and Subject Descriptors," *Hum. Factors*, pp. 97–106.

[47] T. Dybå and T. Dingsøy, "Empirical studies of agile software development: A systematic review," *Inf. Softw. Technol.*, vol. 50, no. 9–10, pp. 833–859, 2008, doi: 10.1016/j.infsof.2008.01.006.

[48] D. Nabil, A. Mosad, and H. A. Hefny, "Web-Based Applications quality factors: A survey and a proposed conceptual model," *Egypt. Informatics J.*, vol. 12, no. 3, pp. 211–217, 2011, doi: 10.1016/j.eij.2011.09.003.

[49] Offutt Jeff, "Quality Attributes of Web Software Applications," *IEEE Softw.*, no. 10.1109/52.991329, pp. 25–32, 2002.

