



**TAYLOR'S
UNIVERSITY**
Wisdom • Integrity • Excellence

**SCHOOL OF
COMPUTING AND IT**

**CAPSTONE PROJECT 1 (BACHELOR OF
SOFTWARE ENGINEERING)**

JMS FOR JESTEC

**FAUZI ANUAR LEE
(0331866)**

School of Computing and IT

JUNE 2019

Supervisor: SUKHMINDER KAUR

ACKNOWLEDGEMENT

The author would like to express his utmost gratitude to Taylor's University for providing the author with the opportunity

Throughout this project....., the author
..... In addition,

.

COPYRIGHT

I do solemnly and sincerely declare that:

- (1) I am the sole author/writer of this Work (except Chapter 1 that is common across the project group);
- (2) This Work is original;
- (3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
- (4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
- (5) I hereby assign all and every right in the copyright to this Work to Taylor's University (TU), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of TU having been first had and obtained;
- (6) I have seemed appropriate permission from the Industry Partner (Client) before including any material(s) that may be deemed as proprietary or sensitive in nature.
- (7) I am fully aware that in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by TU.

Candidate's Signature

Date:

Subscribed and solemnly declared before,

Witness's Signature

Date:

Name:

Designation:

ABSTRACT

Abstract

start

here.....

.....

.....

.....

TABLE OF CONTENTS

TITLE PAGE	i
ACKNOWLEDGEMENT	ii
COPYRIGHT	iii
ABSTRACT	iv
TABLE OF CONTENTS	v
LIST OF TABLES	viii
LIST OF FIGURES	ix
1. Project Description.....	1
1.1 Project Background	1
1.2 Problem Statement	3
1.3 Project Objectives.....	5
1.4 Stakeholders	6
1.4.1 Organization Chart	7
1.5 Project Scope	8
1.6 Risk Management.....	12
1.7 Work Breakdown Structure.....	13
1.8 Activity List.....	14
1.9 Responsibility Assignment Matrix	16
1.10 Project Scheduling	18
2. Literature Review.....	19
2.1 User Interface and User Experience (UI/UX)	19
2.1.1 Clean User Interface.....	19
2.1.2 Visual Flow	20
2.1.3 Building User Trust, Satisfaction and Loyalty.....	21
2.1.4 Usability Guidelines	22

2.2	Implementation Languages	23
2.2.1	HTML5	23
2.2.2	CSS 3.....	24
2.2.3	Ruby	24
2.3	Frameworks & Libraries	25
2.3.1	Bootstrap	25
2.3.2	Ruby on Rails.....	25
2.4	Tools Used for Development	26
2.4.1	GitHub.....	26
3.	System Analysis	27
3.1	Proposed System	27
3.1.1	Website redesign	27
3.1.2	Dashboards	27
3.1.3	Messaging	28
3.1.4	Job Counter	28
3.2	Use Case Diagram	29
3.3	Use Case Specification.....	30
4.	System Design.....	38
4.1	Interface Design	38
4.1.1	Landing Page.....	38
4.1.2	Submit Paper Page	39
4.1.3	Archive Page	40
4.1.4	Journal Article Reader Page.....	41
4.1.5	Login Page	42
4.1.6	Dashboard (Editor).....	43
4.1.7	Dashboard – Search & Filter (Editor).....	44
4.1.8	Dashboard – Job Tracker (Editor).....	45

4.1.9	Dashboard – Tasks (Editor)	46
4.1.10	Dashboard (Reviewer)	47
4.1.11	Dashboard – Job Tracker (Reviewer)	49
4.1.12	Manuscript Reader & Review Form	50
4.1.13	Sending Messages	51
4.1.14	Messages Page	52
4.1.15	Profile Page	53
4.2	Test plan design.....	54
4.2.1	Introduction.....	54
4.2.2	Objective	54
4.2.3	Scope.....	55
4.2.4	Testing Strategy	55
4.2.5	Testing Environment.....	56

LIST OF TABLES

Table 1.1 Project objectives	5
Table 1.2 Stakeholders	6
Table 1.3 Organization Chart.....	7
Table 1.4 Risk Management	12
Table 1.5 Responsibility Assignment Matrix	16
Table 1.6 Project Scheduling	18
Table 3.1 Use Case Specification (Receives Submission).....	30
Table 3.2 Use Case Specification (Checks Paper Format)	31
Table 3.3 Use Case Specification (Assign Associate Editor)	32
Table 3.4 Use Case Specification (Receive Assignment).....	33
Table 3.5 Use Case Specification (Assign Reviewer)	34
Table 3.6 Use Case Specification (Receive Review).....	35
Table 3.7 Use Case Specification (Send Revision Notes)	36
Table 3.8 Use Case Specification (Receive Revision).....	37
Table 4.1 Testing Team Members	54
Table 4.2 Testing Assignment.....	56

LIST OF FIGURES

Figure 3.1 - Use Case Diagram	29
Figure 4.1 Landing Page	38
Figure 4.2 Submit Paper Page.....	39
Figure 4.3 Archive Page.....	40
Figure 4.4 Journal Article Reader Page	41
Figure 4.5 Login Page	42
Figure 4.6 Dashboard (Editor)	43
Figure 4.7 Dashboard - Search & Filter (Editor)	44
Figure 4.8 Dashboard - Job Tracker (Editor)	45
Figure 4.9 Dashboard - Tasks (Editor).....	46
Figure 4.10 Dashboard (Reviewer)	47
Figure 4.11 Dashboard - More Info (Reviewer)	48
Figure 4.12 Job Tracker (Reviewer)	49
Figure 4.13 Manuscript Reader & Review Form.....	50
Figure 4.14 Sending Messages.....	51
Figure 4.15 Messages Page	52
Figure 4.16 Profile Page.....	53

1. Project Description

1.1 Project Background

Our team is tasked with the development of a Journal Management System (JMS) for Taylors university research journal JESTEC. We aim to implement a web cloud-based solution that will be replacing the current system. Features to be included and commercialized are the submission, review, and acceptance of journal/manuscripts online with automated cloud support. The expected results are to increase the traffic and revenue by speeding up the journal submission/review process. Our goal is to deploy a journal submission system that is completely online using web and cloud services. The process takes place as an author that's trying to submit their research paper, although before doing so, they'll be greeted with the requirements and the format that the paper needs to be in before it goes through the reviewing phase. Even so, the author would be expected to already have made an account on the website, if not, then it'll be instructed to them, the website aims to be more friendly towards the user based on its simplicity and it's a step by step guide on showing you how to submit your research paper into the review process. The guidelines consist of its figures, equations, tables, nomenclature, and the limit of the paper itself. After that is done, the editor will receive a notification from the website notifying the submission of a paper that has taken place, and it will require the attention of the editor within the week. Assuming the editor checks the submission of the paper straight away, the paper will then be assigned to an associate editor, as he's the one responsible for picking out the reviewers, the reviewers will have 4 weeks to review the paper if they choose to accept review process. Reviewers will receive an email for the request to review. After the review is done, it'll be sent back to the associate editor on the comments regarding the paper if there is any, if revision is needed, an email will be sent to the author by the associate editor, regarding the faults that were made in the paper, and needs to be reworked, the author has 4 weeks to work on it when everything is done, the associate editor will have the final decision to accept the paper or to reject, hence the revision. Once the paper has been submitted, the associate editor will send the paperback to the editor with formatting the paper to the current standard of the website, and the editor

will request the author to get a copyright for the research paper to be submitted to the website.

JMS will be developed similar to JESTEC's current system, as we want to make sure that current users will easily transition to the new system, but on the backend part of the system it'll be completely different as we've said before, it'll be a cloud-based system that automates most of the work that previously one of the stakeholders has the responsibilities of doing so, things like storing elements of the website, for example, the submitted research papers, will be stored in the cloud. Unlike now, it is stored on one of the stakeholder's machines, which we find not so easy for all the 4 clients, as all the files will be going through him first. Other than that, we also aim to remove the clutter that's existent on the website, if you head to JESTEC, there will be a lot of information served at your face, and we think that it might be demotivating for some people, that's why we want our website to be user friendly, guiding the user step by step on the process of submitting a paper, simplicity will be a big factor in JMS.

1.2 Problem Statement

The current JMS process is handled manually, including all communication. From the moment of manuscript submission, the review process can take well over a month to complete a result. During this, there is communication between multiple parties necessary.

Thus, we can identify the main issues of the manual process to:

- Communication between parties
- Record of submissions and their status
- Lack of automation (Forms, messaging, etc.)
- Tracking of workflow
- Search and filter information (reviewers, submissions, etc.)

Validation and reliability of data could be compromised since processes are not automated, human errors are inevitable. Without the ability to validate automatically an error could lead to delay of the entire process. Especially with the recurring communication between multiple parties.

Automation allows the system to minimize the input needed by the user, thus reducing human error. Additionally, it would allow information to be stored effectively, allowing for validation as well as the reliability of the data collected.

The login system for people involved is completely missing which does not allow for validation of the user, furthermore, processes must be tracked by the individual themselves. Without the knowledge of the workflow, this can prove to be a challenge and prone to additional errors.

Communication is currently handled entirely through email. With deadlines constantly having to be met it tracking and sending emails to individuals can be challenging. Email addresses of the individuals involved must be kept track of manually. Lastly, an issue with the use of personal email is the clutter in the mailbox, which does not allow for specialized categorization for JMS purposes.

The ability to **search and filter** for information would improve the time needed for the user to find or see what they need to see. This would help increase the speed and

ease of use of the system. Reducing the visual clutter to what is most important to the user at the time.

Due to the nature of the current system being fully manual, the manual storage of information does not allow for analytics to be run on all the data sets. Analytics can allow the user to gain further insight and reveal information previously unclear.

The current website does not adhere to the web design principles [6]; information is scattered throughout the website, causing the website to look cluttered and unpleasant to look at. This also makes it harder for users to navigate through the website and find the information that they are looking for

1.3 Project Objectives

Table 1.1 Project objectives

No.	Proposed Functionality	Problems Solved/ Opportunities
1	Website redesign	Web design principles[6]
2	User Login/ Sign-up	Login system[3]
3	Author functionality	Automation[2], Search and filter[5]
4	Editor functionality	Automation[2], Search and filter[5]
5	Reviewer functionality	Automation[2], Search and filter[5]
6	System admin functionality	Automation[2]
7	Messaging	Communication[4]
8	Template management	Automation[2]
9	Workflow/ Job counter	Automation[2]
10	Storage of data	Validation and reliability of data[1], Automation[2]
11	Threat Modelling	Security issues
12	Penetration Testing	Security issues
13	Unit Testing	Validate that each unit of the software performs as designed
14	Integration Testing	Validate that individual units of the software performs well when joined as a group
15	System Testing	Validates the complete and fully integrated software product
16	Acceptance Testing	Validate the system's compliance with the business requirements and determine whether it is acceptable for launching

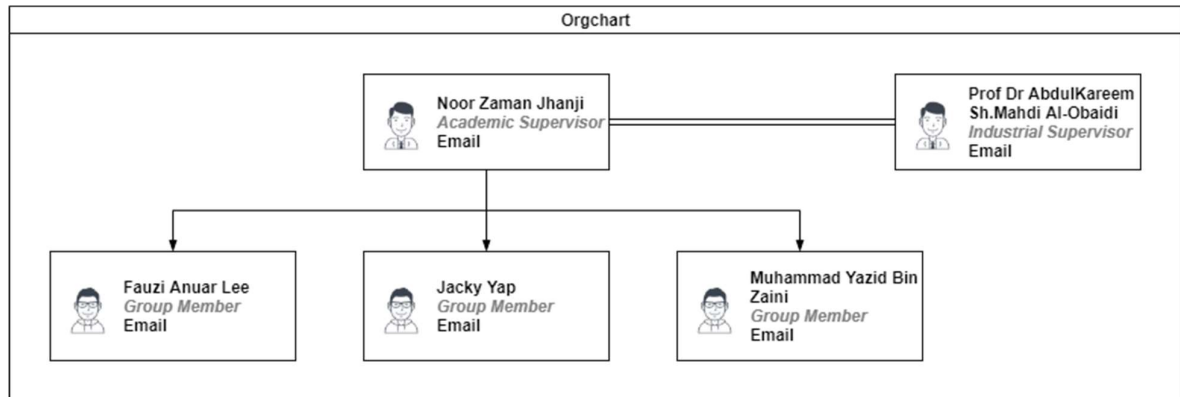
1.4 Stakeholders

Table 1.2 Stakeholders

Stakeholder	Role(s)	Responsibilities
Prof Dr AbdulKareem Sh.Mahdi Al- Obaidi	Project Sponsor	To provide information on the scope, monitoring the progress, and providing any other assistance that is required.
Noor Zaman Jhanjhi	Project Supervisor	To provide guidance and support in term of knowledge, opinion and view throughout the entire project. Furthermore, will be evaluating the project and provide recommendation to continuously improve the project.
Fauzi Anuar Lee	<ul style="list-style-type: none"> • UI/UX Designer • System Analyst • Programmer • Tester 	<p>Lead UI/UX designer. Analyze the visual hierarchy of the website and research UI elements of existing systems.</p> <p>To analyze and research on system requirements.</p> <p>To implement the proposed solutions into functional software.</p> <p>To test and validate the system.</p>
Jacky Yap	<ul style="list-style-type: none"> • Data Engineer • System Analyst • Programmer • Tester 	<p>Design and implement the data base schematics.</p> <p>To analyze and research on system requirements.</p> <p>To implement the proposed solutions into functional software.</p> <p>To test and validate the system.</p>

1.4.1 Organization Chart

Table 1.3 Organization Chart



1.5 Project Scope

The JMS will be developed for use on a web platform with cloud solutions. The 5 main users will be able to access all functions respectively through the website. The 5 primary users are:

- 1. Editors**
 - a. Receive manuscripts
 - b. Assign manuscripts to Associate Editors
- 2. Associate Editors**
 - a. Assign reviewers
 - b. Send reminders
 - c. Approve or deny manuscripts
- 3. Authors**
 - a. Submit manuscripts
 - b. Nominate reviewers
 - c. Receive review feedback
 - d. Submit copyright transfer form
- 4. Reviewers**
 - a. Review manuscripts
- 5. System Admins**
 - a. Publish approved manuscripts on website

The last users to be considered are readers that visit the website to browse through the publications.

The creation of this solution aims to provide a platform for journals to be reviewed, published and read. In order to deliver a great experience for all types of users, we will streamline the workflow and automate numerous tasks. For our JMS, we want it to be intuitive to use, without the need for extensive training. In order to achieve this, we reduce visual clutter and provide a clear sequence in workflow and bring attention to tasks that immediate feedback.

Database

To create a database system that suits the needs for this JMS. A database will allow the storage of all information regarding the JMS. It would store data that will be used in the website such as the journals published as well as reviewer's information. A database allows for easy storage of information from automated forms as well as the access to it through queries and other means.

Security system

The latest security system will be applied into the website to prevent malicious parties from accessing sensitive data stored in the database. For instance, to ensure only the authorized users can access and edit the database, each user will be given different levels of access. The website application will also use two-factor authentication (2FA) to further secure the system.

Web platform

Will provide an environment for the JMS and its users. We have created a sitemap to showcase the hierarchical order of the webpage. An added benefit of creating the sitemap is to enable Google search engine crawlers to index our website and its content; this allows our website to appear on their search engine and be found easily.

Landing Page

Redirects user to pages that display information about

- How to submit a paper
- Index & awards
- List of editors
- List of reviewers
- Archives of past issues

Dashboard

The Dashboard is the page where most of the system's functions are accessed

- Information about the manuscripts involved with the user is displayed here. The information displayed on this page varies from user to user (author, reviewer, editor).
- Among the most important information displayed here to the user is the manuscript status.
- Users can filter their manuscripts according to status.
- By clicking on the desired manuscript, a pop-up will appear where they are able to perform various tasks concerning the manuscript.

Manuscript Pop-up

The functions on this pop-up vary from user to user (author, reviewer, editor). Editors can:

- Contact author(s)
- Assign editor
- Assign/add reviewer
- Change status
- Send message
- Read paper
- Review paper

Whereas reviewers are only able to:

- Send message
- Read paper
- Review paper

Send Message

From the pop-up page, users can send messages to relevant parties involved with the manuscript.

- Users can send various pre-determined messages to other parties (authors, editors, reviewers).
- Information stored in the database will be used to fill in the blanks in these messages.
- With this, users will not have to take the time to draft a message every time.
- Only editors can edit their messages before sending them.

Messages Page

Messages are compiled based on their respective manuscripts. This way, users can easily keep track of the messages they have received about them.

1.6 Risk Management

Table 1.4 Risk Management

Risks	Impact Level	Counter Measurements
Client wants to change design of website	Low	<ul style="list-style-type: none">- Make sure that the code foundation of the website is easy to be updated or replaced- Make sure that the client is satisfied with the design before we start developing
Inexperienced team in cloud computing	Medium	-The lack of experience in indulging ourselves in cloud computing can be quite risky for the project, but we believe in the platform we have chosen to make it as easy as possible for us to transition into the cloud computing era.
Time given for the project	Medium	<ul style="list-style-type: none">-We are requesting an opportunity from Taylor's to let us intern there while working on the project with the budget required to further improve the quality of the project.-If that does not work out, we plan to start working on it as soon as possible while having applications like Trello and Google Meet for making sure everything is going as planned.
MCO Lockdown Extensions	Low	<ul style="list-style-type: none">-If the world is back in shock with numbers rising, we will decide to use Google Meet for frequent meetings on the advancements of the project.-Productivity overall will be lower as we can't have physical meetings and the ability to experiment things in the computer lab will be unavailable.

1.7 Work Breakdown Structure

1. Project Initiation

- 1.1. Project Background
- 1.2. Problem Statement/Opportunities
- 1.3. Project Objectives
- 1.4. Stakeholders
- 1.5. Project Scope
- 1.6. Risk Management
- 1.7. Work Breakdown Structure
- 1.8. Activity List
- 1.9. Responsibility Assignment Matrix
- 1.10. Project Scheduling

2. Literature Review

- 2.1. Technology
- 2.2. Cloud Service Providers
- 2.3. Domain Research

3. System Analysis

- 3.1. Current System Analysis
- 3.2. Dataflow Diagram
- 3.3. Proposed System
- 3.4. Use Case Diagram
- 3.5. Use Case Specification
- 3.6. SWOT Analysis
- 3.7. Workflow diagrams of existing system

4. System Design

- 4.1. Interface Design
- 4.2. Database Design
- 4.3. Workflow of proposed system
- 4.4. Class diagram
- 4.6. Design sequence diagram
- 4.7. Test plan design
- 4.9. Pseudocode

1.8 Activity List

1. Project Initiation

- 1.1. Project Background*
- 1.2. Problem Statement/Opportunities*
- 1.3. Project Objectives*
- 1.4. Stakeholders*
- 1.5. Project Scope*
- 1.6. Risk Management*
- 1.7. Work Breakdown Structure*
- 1.8. Activity List*
- 1.9. Responsibility Assignment Matrix*
- 1.10. Project Scheduling*

2. Literature Review

2.1. Technology

- 2.1.1. Implementation Language*
 - 2.1.1.1. HTML*
 - 2.1.1.2. CSS*
 - 2.1.1.3. Ruby*
 - 2.1.1.4. MySQL*
- 2.1.2. Tools*
 - 2.1.2.1. Ruby on Rails*
 - 2.1.2.2. Bootstrap*
 - 2.1.2.3. Microsoft Security Graph*
 - 2.1.2.3. MySQL Security*
 - 2.1.2.4. GitHub*

2.2. Cloud Service Providers

- 2.2.1. Microsoft Azure*
- 2.2.2. Amazon Web Services (AWS)*
- 2.2.3. Google App Engine*
- 2.2.3. Price comparison*

2.3. Domain Research

- 2.3.1. Analyse peer-review process*

- 2.3.1. *Analyse previous security model*
- 2.3.1. *Identify security solutions*
- 2.3.2. *Case Studies*
- 2.3.3. *Web design principles & usability*

3. System Analysis

3.1. Current System Analysis

3.2 Dataflow Diagram

3.3. Proposed System

3.4. Use Case Diagram

3.5. Use Case Specification

3.6. SWOT Analysis

3.7. Workflow diagrams of existing system

4. System Design

4.1. Interface Design

- 4.1.1. *Archive Page*
- 4.1.2. *Journal Article Reader Page*
- 4.1.3. *Dashboards*
- 4.1.4. *Manuscript Reader & Review Form*
- 4.1.5. *Messages Feature*

4.2 Database Design

- 4.2.1. *Entity Relationship Diagram (ERD)*
- 4.2.2. *Data Dictionary*
- 4.2.3. *MySQL Implementation*

4.3. Workflow of proposed system

4.4. Class diagram

4.6. Design sequence diagram

4.7. Test plan design

4.9. Pseudocode

1.9 Responsibility Assignment Matrix

Table 1.5 Responsibility Assignment Matrix

Responsibility	Fauzi	Jacky	Yazid
1. Project Initiation			
1.1. Project Background	✓	✓	✓
1.2. Problem Statement/Opportunities	✓	✓	✓
1.3. Project Objectives	✓	✓	✓
1.4. Stakeholders	✓	✓	✓
1.5. Project Scope	✓	✓	✓
1.6. Risk Management	✓	✓	✓
1.7. Work Breakdown Structure	✓	✓	✓
1.8. Activity List	✓	✓	✓
1.9. Responsibility Assignment Matrix	✓	✓	✓
1.10. Project Scheduling	✓	✓	✓
2. Literature Review			
2.1. Technology			
2.1.1. Implementation Language			
2.1.1.1. HTML	✓		
2.1.1.2. CSS	✓		
2.1.1.3. Ruby	✓		
2.1.1.4. MySQL		✓	✓
2.1.2. Tools			
2.1.2.1. Ruby on Rails	✓		
2.1.2.2. Bootstrap	✓		
2.1.2.3. Microsoft Security Graph			✓
2.1.2.3. MySQL Security			✓
2.1.2.4. GitHub	✓		
2.2. Cloud Service Providers			
2.2.1. Microsoft Azure		✓	✓
2.2.2. Amazon Web Services (AWS)		✓	✓
2.2.3. Google App Engine		✓	
2.2.3. Price comparison		✓	
2.3. Domain Research			
2.3.1. Analyze peer-review process		✓	
2.3.1. Analyze previous security model			✓
2.3.1. Identify security solutions			✓
2.3.2. Case Studies		✓	✓
2.3.3. Web design principles & usability	✓		

3. System Analysis			
3.1. Current System Analysis		✓	
3.2. Dataflow Diagram			✓
3.3. Proposed System	✓		
3.4. Use Case Diagram	✓	✓	
3.5. Use Case Specification	✓	✓	
3.6. SWOT Analysis			✓
3.7. Workflow diagrams of existing system			✓
4. System Design			
4.1. Interface Design			
4.1.1. Archive Page	✓		
4.1.2. Journal Article Reader Page	✓		
4.1.3. Dashboards	✓		
4.1.4. Manuscript Reader & Review Form	✓		
4.1.5. Messages Feature	✓		
4.2 Database Design			
4.2.1. Entity Relationship Diagram (ERD)		✓	
4.2.2. Data Dictionary		✓	
4.2.3. MySQL Implementation		✓	
4.3. Workflow of proposed system			✓
4.4. Class diagram			✓
4.6. Design sequence diagram			✓
4.7. Test plan design	✓		
4.9. Pseudocode			✓

1.10 Project Scheduling

Table 1.6 Project Scheduling

No.	Activities	Days	Start Date	Finish Data
1.	Requirement Gathering	7	Week 1	Week 2
2.	List objectives	7	Week 2	Week 3
3.	Survey old website	14	Week 1	Week 3
4.	Client comment specification	14	Week 1	Week 3
5.	Threat Modelling	7	Week 3	Week 4
6.	Client review specification	60	Week 1	Week 9
7.	Prototyping	60	Week 1	Week 9
8.	Domain Research	7	Week 3	Week 4
9.	Implementation Language Research	7	Week 3	Week 4
10.	Research on Tools needed	7	Week 3	Week 4
11.	Cloud service provider needed	7	Week 3	Week 4
12.	Design the dataflow of the project	7	Week 4	Week 5
13.	Create proposed system	7	Week 4	Week 5
14.	Specify use case scenario	7	Week 4	Week 5
15.	Analyse the SWOT	7	Week 5	Week 6
16.	UI/UX designing	60	Week 4	Week 12
17.	Database designing	14	Week 4	Week 6
18.	Creating the ERD	14	Week 5	Week 7
19.	Creating the Design sequence Diagram	7	Week 3	Week 4
20.	Analysing the workflow of proposed system	7	Week 3	Week 4
21.	Creating class diagram	7	Week 10	Week 11
22.	Creating the test plan design	7	Week 7	Week 8
23.	Listing out the pseudocode	7	Week 8	Week 9

2. Literature Review

2.1 User Interface and User Experience (UI/UX)

According to Lindgaard et al. (2006), users will form their impressions of a website in just 50 milliseconds upon visiting it. This crucial time is what decides whether the user will stay on the website and whether they will visit again. Keeping this in mind, the importance of having good user interface and user experience (UI/UX) becomes apparent; to retain visitors in a website and preventing them from ‘bouncing’. A bounce is when a visitor views a single page in your website and leaves (Google, 2020). In order to reduce the bounce rate of the website, we must consider the factors that allow visitors to have a good first impression of the website. This literature review discusses the factors that should be considered when developing the user interface of the website to ensure that the website is user-friendly.

2.1.1 Clean User Interface

Oyibo and Vassileva (2017) states that classical aesthetics is stronger than expressive aesthetics in determining the degree of perceived usability of a website. In Human-Computer Interaction (HCI) design constructs, classical aesthetics relates to ‘clean’ and ‘orderly’ design whereas expressive aesthetics relates to ‘creative’ and ‘fascinating’ design. The study made by Oyibo and Vassileva has shown that visitors tend to associate clean user interfaces with a high degree of usability. Thus, in our project, we hope to improve the website’s usability by reducing the visual clutter and come up with a clean user interface that is easy on the eyes.

2.1.2 Visual Flow

Another factor to consider is the visual flow of the website. One can design a website in such a way that they can direct the user's gaze along a give path based on the principles of visual hierarchy. Pang et al. (2016) has proposed a novel technique that allows web designers to achieve this through gathering and analyzing eye-tacking data from real-world users on various existing webpages. This technique serves to predict the path of the user's attention and the order of which they look through the components of a web page. Taking inspiration from this study, we are able to apply this technique to our website and improve its visual flow in order to direct the user to the main features of the website. This also serves as a deterrent that will help us avoid cluttering key pages of the website with unnecessary information that may cause the visitors to bounce. However, eye-tracking technology may be too costly, and it is undecided whether such resources would be allocated for this project.

As an alternative, we could utilize mouse movement data to observer the user's visual flow instead. Clark and Stephane (2018) suggests that eye gaze and mouse cursor position can strongly correlate. In order to further investigate this, they have conducted a study that uses *The Eye Tribe Eye Tracker* and *EyeProof* to find correlation between eye gaze and mouse position on the Florida Institute of Technology Human-Centered Design Institute website. Results indicate that mouse tracking data could be an effective alternative for eye tracking data on the studied website and it may be feasible to use it to conduct similar assessments. Thus, we will be using mouse movement data instead of eye-tracking data when using Pang et al.'s novel technique to analyze the user's user flow.

2.1.3 Building User Trust, Satisfaction and Loyalty

Faisal et al. (2017) had referenced one of Hofstede's cultural dimensions which is the uncertainty avoidance. Uncertainty avoidance can be defined as 'the extent of which a community avoids unknown situations and ambiguity' (Hofstede, 2001). This dimension shows that visitors would be very likely to avoid websites that would make them feel uncertain as to whether the contents of the website are credible, hence would cause the website's bounce rate to increase. Faisal et al. has concluded in their study that content quality and navigation were important factors in building user trust in a website whereas interactivity, color and typography were crucial in ensuring user satisfaction. As mentioned earlier in this review, it is crucial that visitors form a good impression lest they bounce from the website; the study conducted by Faisal et al. further enforces this. By prioritizing on factors that would build user trust and loyalty, we are able to circumvent uncertainty avoidance culture amongst visitors, furthermore, improving the website's visitor retention. To apply this to our project, we will be taking the factors above into consideration when designing the website.

2.1.4 Usability Guidelines

Nielsen (1994) has developed a set of broad, yet reliable rules of thumb called *Jakob Nielsen's 10 Usability Heuristics for User Interface Design* that is widely used amongst UI developers to this day. However, Alonso-Virgós et al. (2020) suggests that this set of heuristics is too broad and despite its contributions, experts in UX have shown that these heuristics are subpar solutions to design problems due to their theoretical nature (Lodhi, 2010). In response to this, Alonso-Virgós et al. extracted, grouped and classified a standardized list of useful recommendations applicable to all websites:

- a. Respect conventional layout*
- b. Uniformity and consistency in the design*
- c. Avoid "false clickable elements"*
- d. Common properties for the search bar*
- e. Respect the common order of the buttons*
- f. Respect the common order of the elements of a form*
- g. Proper use of colours*
- h. Proper use of common icons*

In addition to Nielsen's Usability Heuristics, we plan to use the recommendations proposed by Alonso-Virgós et al. to further improve the usability of our website and ensure that it is user-friendly.

2.2 Implementation Languages

2.2.1 *HTML5*

Hypertext Markup Language (HTML) is the standard markup language for websites developed by the World Wide Web Consortium (W3C) and is considered to be the most basic building block of the Web (Mozilla, 2020). The latest version of the standard that defines this markup language is HTML5. Since the release of this version, the previous versions of HTML are considered obsolete, thus it is a given that we will be using this version of HTML instead. On top of that, this version comes with many new additions to the markup language (Mozilla, 2019). Among them are:

- **Semantics:** allows you to describe your content with more precision.
- **Connectivity:** allows communication with the server in new and innovative ways.
- **Offline and storage:** allows websites to operate offline more efficiently and store data on the client-side locally.
- **Multimedia:** enables embedding audio and video into websites more easily.
- **2D/3D graphics and effects:** allows a diverse range of presentation methods.
- **Performance and integration:** provide faster optimizations and efficient usage of computer hardware.
- **Device access:** allows for the usage of various input and output devices.
- **Styling:** lets authors design more intricate themes

2.2.2 CSS 3

Cascading Style Sheets (CSS) is a stylesheet language that is used alongside HTML as a way to describe the presentation of the elements written in HTML; for instance: font family, color and size (Mozilla, 2020). This language will be essential in ensuring that our website would possess an aesthetic look and feel. There are several versions of CSS such as CSS1, CSS2.1, CSS3 and CSS4, though CSS4 was never considered to be an official version. Hence, we will be using CSS3 in our project since it is currently the most stable version.

2.2.3 Ruby

Ruby is an object-oriented programming language that boasts itself a high-level programming language that is easy to use, yet practical (Ruby, n.d.). Everything involved in Ruby is an object; any given data type comes with their own methods and instance variables. Ruby also features Blocks, which allows the developer to attach a closure to any method, hence offering a great amount of flexibility. Most of all, this programming language has a great amount of resources and references posted online by the community, which would help us greatly if and when we are stuck with an issue. With how simple yet powerful this programming language is, we have decided to use Ruby as our back-end programming language instead of JavaScript and PHP.

2.3 Frameworks & Libraries

2.3.1 *Bootstrap*

Bootstrap is the most commonly used free and open-source CSS Framework that features prebuilt components, themes and JavaScript plugins (Bootstrap, n.d.). Using Bootstrap in our project would ensure that our website is responsive, no matter what resolution the user's device is using. Among some of the resources available in this framework are Bootstrap Icons; Bootstrap's very own open source SVG icon library with all the icons that we would possibly need. In the case that we are not allocated funds for the development of the website, we are also able to use resources from Start Bootstrap, where they offer a number of Bootstrap themes, templates and snippets for free (Start Bootstrap, n.d.). They also have a beginner friendly guide on how to publish websites using their themes and GitHub Pages.

2.3.2 *Ruby on Rails*

Rails is a web application development Rails is a platform for creating web applications, written in the Ruby programming language (Rails, n.d.). It's designed to make web application programming simpler by making assumptions about what each developer needs to get started. It allows you to write less code while doing more than many other languages and frameworks. Experienced developers of Rails also report making development of web applications more amusing.

Rails is a software that is opinionated in nature. It makes the presumption that there's a "right" way of doing things, and it's structured to promote that way; and in some cases, to discourage alternatives. When you know "The Rails Way" you will possibly discover a huge productivity increase. If you persist in bringing old habits from other languages into the development of your Rails, and try to use patterns you have learned elsewhere, you're going to have a bad time.

We've decided to include Rails in our project due to its rising popularity in the web design scene. By including it into our project, we will have the opportunity to learn this framework whilst working with it.

2.4 Tools Used for Development

2.4.1 GitHub

Git is a version control system that enables us to manage our revisions as we will be making constant changes to the codes and releasing new versions throughout development. Our code will be pushed into the central repository of Git, thus enabling us to collaborate easily as we can just download the latest version of the software, make some changes and push the newest revision (Brown, 2019).

GitHub, as an extension of Git, is where developers around the world store their projects and form connections. Through this, we are able to reference from online sources in GitHub when we need to look something up during development.

3. System Analysis

3.1 Proposed System

The proposed system will be built on a cloud solution which contains all previous functions of the current system with additional features and enhancements to further facilitate the end user's experience. This section will introduce the various features and enhancements that will be included in our system to address the problem statements identified.

3.1.1 Website redesign

The problem with the current system is that the website has an outdated design and has sub-par usability that could be enhanced by revamping the web design of the system to a much more modern and cleaner look and feel. To achieve this, we will be reducing the visual clutter in the website and minimise the number of clicks for users to find the information they require to ensure that the website is intuitive for all users. We will also be using a very neutral colour palette during the website's redesign to reduce eye strain in consideration of the users who are likely to use this website for long periods of time.

3.1.2 Dashboards

The current system does not have an existing dashboard feature. Among other problems with the current system, the lack of automation and difficulty of retrieving information are some problems that could be addressed by implementing a dashboard feature into the system. By having a space where users are able to perform all their tasks in one place, the usability of the system increases exponentially. In this dashboard, information is automatically sorted so that users will have an easier time finding the information that they require. On top of that, a search and filter function will be implemented into the dashboard to further assist the user in information retrieval.

3.1.3 Messaging

The current process relies on third party applications (such as Gmail) to communicate between users. This makes it necessary for users to manage their inboxes and messages frequently, which is an unnecessary hassle that they need to go through. If the messages are not managed properly, it could get very confusing for the user and amidst the confusion, users may miss out on very crucial information. Due to this, we have implemented a built-in communication system that categorises the user's messages according to the submissions they're involved in. This will neatly organise the user's messages into easy-to-find collections that facilitates the review process and greatly reduce the possibility of miscommunication.

3.1.4 Job Counter

The current system does not have an existing job counter feature. One of the problems identified with the current system is that users may lose track of the many tasks that they need to perform. Users will need to personally keep track of the tasks that they need to do, and if the user has many tasks ongoing, they could feel overwhelmed. We've decided to include a job counter feature into the system to address this issue. By implementing it in the user's dashboard, the tasks that the users need to perform will be displayed and sorted by nearest due date. This enables the user to make educated decisions in selecting which jobs to prioritise and remind them when a task is approaching its due date.

3.2 Use Case Diagram

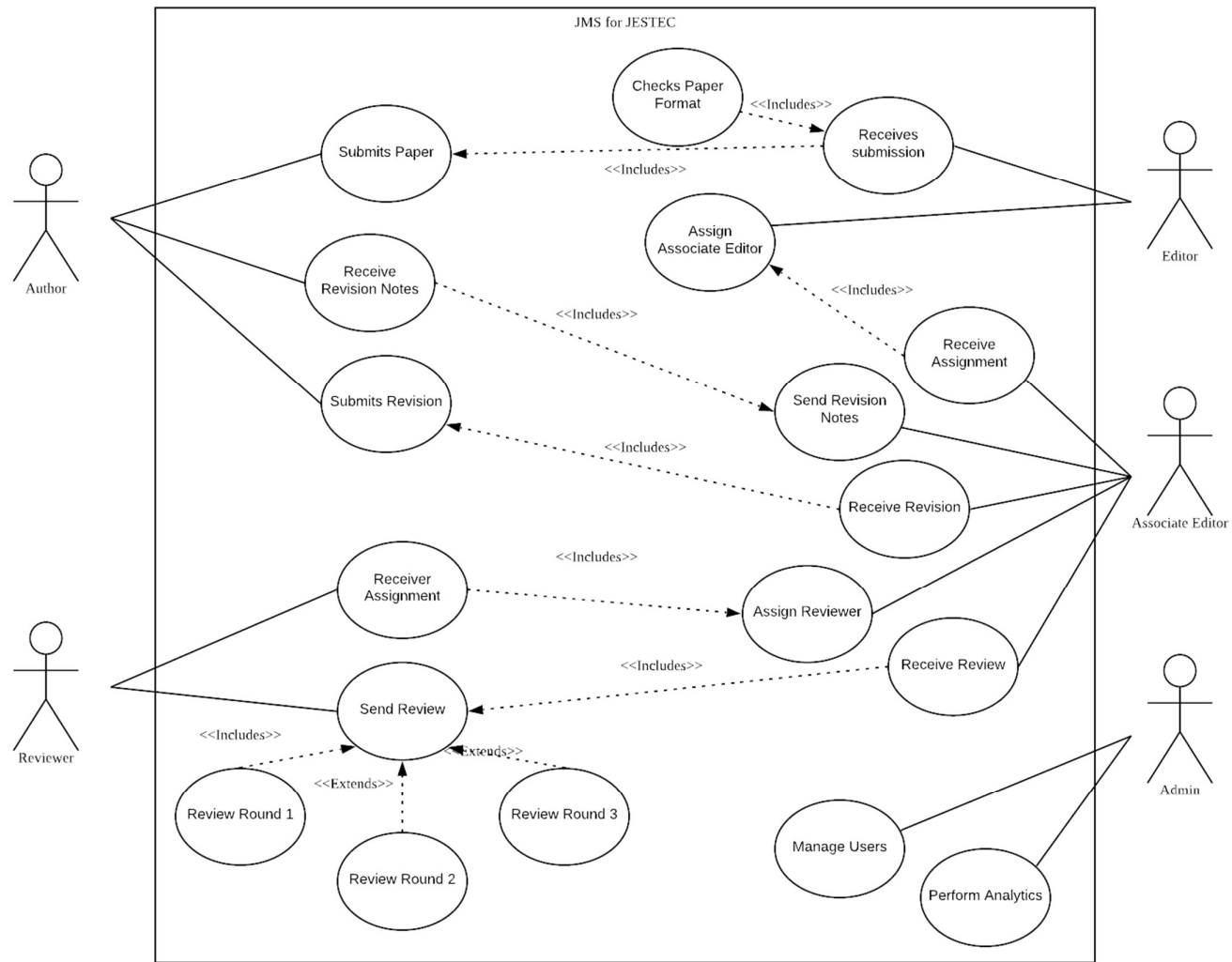


Figure 3.1 - Use Case Diagram

3.3 Use Case Specification

Table 3.1 Use Case Specification (Receives Submission)

Use Case Name	Receives Submission
Actor(s):	Editor (primary), Author (secondary)
Summary Description:	Allows editors to receive manuscript submission from author
Pre-Condition:	The author has sent their manuscript submission
Post-Condition(s):	<ul style="list-style-type: none">• Editor checks manuscript format• Editor accepts/declines submission
Basic Path:	<ol style="list-style-type: none">1. Author submits manuscript to Editor2. The editor checks whether the manuscript format adheres to JESTEC guidelines3. Editor accepts the manuscript submission4. Editor assigns the submission to an associate editor to handle
Alternative Paths:	<ol style="list-style-type: none">1. Author submits manuscript to Editor2. The editor checks whether the manuscript format adheres to JESTEC guidelines3. Editor declines the manuscript submission4. Editor notifies the author that their submission is declined due to nonadherence to JESTEC guidelines.

Table 3.2 Use Case Specification (Checks Paper Format)

Use Case Name	Checks Paper Format
Actor(s):	Editor (primary)
Summary Description:	Allows editors to check the manuscript whether it adheres to JESTEC guidelines.
Pre-Condition:	The editor has received the manuscript submission.
Post-Condition(s):	Editor approves the format of the manuscript
Basic Path:	<ol style="list-style-type: none"> 1. Editor skims through the manuscript 2. Editor determines that the manuscript is acceptable 3. Editor proceeds to assign an associate editor
Alternative Paths:	<ol style="list-style-type: none"> 1. Editor skims through the manuscript 2. Editor determines that the manuscript is unacceptable 3. Editor notifies the author that their submission is declined due to nonadherence to JESTEC guidelines.

Table 3.3 Use Case Specification (Assign Associate Editor)

Use Case Name	Assign Associate Editor
Actor(s):	Editor (primary), Associate Editor (secondary)
Summary Description:	Allows editors to assign manuscript submissions for associate editors to handle.
Pre-Condition:	The editor has approved the manuscript submission.
Post-Condition(s):	Editor notifies the associate editor regarding their assignment
Basic Path:	<ol style="list-style-type: none"> 1. Editor goes through a list of associate editors 2. Editor selects an associate editor 3. Editor sends the assignment to selected associate editor 4. Associate editor accepts the assignment
Alternative Paths:	<ol style="list-style-type: none"> 1. Editor goes through a list of associate editors 2. Editor selects an associate editor 3. Editor sends the assignment to selected associate editor 4. <ol style="list-style-type: none"> a. Associate editor declines the assignment b. No associate editors are available

Table 3.4 Use Case Specification (Receive Assignment)

Use Case Name	Receive Assignment
Actor(s):	Associate Editor (primary), Editor (secondary)
Summary Description:	Allows associate editors to receive submission assignments for them to handle.
Pre-Condition:	The editor has assigned the associate editor to assign the submission
Post-Condition(s):	Editor accepts and assumes responsibility of the assignment
Basic Path:	<ol style="list-style-type: none">1. Editor assigns associate editor to handle a submission2. Editor receives the assignment and assumes responsibility of the assignment
Alternative Paths:	<ol style="list-style-type: none">1. Editor assigns associate editor to handle a submission2. Associate editor declines receiving the assignment3. Editor reassigns submission to another associate editor

Table 3.5 Use Case Specification (Assign Reviewer)

Use Case Name	Assign Reviewer
Actor(s):	Associate Editor (primary), Reviewer (secondary)
Summary Description:	Allows associate editors to send a review request to a reviewer.
Pre-Condition:	The associate editor is assigned the responsibility of handling the submission
Post-Condition(s):	Reviewer accepts review request
Basic Path:	<ol style="list-style-type: none">1. Editor sends a review request to the reviewer2. Reviewer accepts the review request
Alternative Paths:	<ol style="list-style-type: none">1. Editor sends a review request to the reviewer2. Reviewer declines the review request

Table 3.6 Use Case Specification (Receive Review)

Use Case Name	Receive Review
Actor(s):	Associate Editor (primary), Reviewer (secondary)
Summary Description:	Allows associate editors to receive a review from the reviewer.
Pre-Condition:	The reviewer sends their review of the manuscript to the associate reviewer
Post-Condition(s):	Associate editor receives the review written by the reviewer
Basic Path:	<ol style="list-style-type: none"> 1. Reviewer sends a review to the associate editor within 4 months of accepting the review request 2. Associate editor receives the review
Alternative Paths:	<ol style="list-style-type: none"> 1. Reviewer does not send their review before the deadline 2. Associate editor sends a reminder to the reviewer 3. Associate editor extends the deadline of the submission of review

Table 3.7 Use Case Specification (Send Revision Notes)

Use Case Name	Send Revision Notes
Actor(s):	Associate Editor (primary), Author (secondary)
Summary Description:	Allows associate editors to send the revision notes provided by the reviewer to the author.
Pre-Condition:	The associate editor has received the revision notes from the reviewer
Post-Condition(s):	Author receives the revision notes from the associate editor.
Basic Path:	<ol style="list-style-type: none"> 1. Associate editor receives review from reviewer 2. Associate editor summarises the review and compiles the revision notes 3. Associate editor notifies the author that revisions are needed and sends the revision notes to the author
Alternative Paths:	<ol style="list-style-type: none"> 1. Associate editor receives review from reviewer 2. Associate editor summarises the review and compiles the revision notes 3. Associate editor notifies the author that major revisions are needed and rejects the author's submission.

Table 3.8 Use Case Specification (Receive Revision)

Use Case Name	Receive Revision
Actor(s):	Associate Editor (primary), Author (secondary)
Summary Description:	Allows associate editors to receive the revised manuscript from the author.
Pre-Condition:	The author has sent the revised manuscript to the associate editor.
Post-Condition(s):	Associate editor approves the manuscript revision for publishing
Basic Path:	<ol style="list-style-type: none"> 1. Associate editor receives revision from author 2. Associate editor sends the revised manuscript to the reviewer for 2nd round of review 3. Reviewer acknowledges the revision and gives the associate editor their approval 4. Associate reviewer approves the manuscript for publishing.
Alternative Paths:	<ol style="list-style-type: none"> 1. Associate editor receives revision from author 2. Associate editor sends the revised manuscript to the reviewer for 2nd round of review 3. a. Reviewer does not acknowledge the revision and sends in another set of revision notes (2nd round) 4. a. Associate editor notifies the authors that their manuscript requires revision and sends the revision notes (2nd round) 3. b. Reviewer does not acknowledge the revision (3rd round) 4. b. Associate reviewer notifies the authors that their manuscript was rejected. (3rd round)

4. System Design

4.1 Interface Design

4.1.1 Landing Page



Figure 4.1 Landing Page

The landing page is an important part that makes or breaks a website. It takes only a fraction of a minute for visitors to decide whether they like this website or not, and it is important to build a good first impression to make sure the visitors would come again.

With this in mind, we have decided to have a clean interface with minimal information to avoid visual clutter. We have also added some ‘call-to-action’ buttons to generate traffic into the more relevant pages such as submitting a paper and learning more about JESTEC.

As an unregistered user, one is granted access to all the pages in the menu bar, as it is open to the public. However, to access the dashboard and main functions of the website, the user needs to be registered either as an author, reviewer or editor.

4.1.2 Submit Paper Page



Figure 4.2 Submit Paper Page

Before the authors can submit their paper, they need to be aware of the procedures and guidelines set by JESTEC on how to submit their paper and what format their paper should be written in. To achieve this, we've linked the 'Submit Paper' button to a page that documents these procedures and guidelines.

Additionally, the users need to be signed in to submit a paper, so we added a login/signup button at the bottom of this page

4.1.3 Archive Page

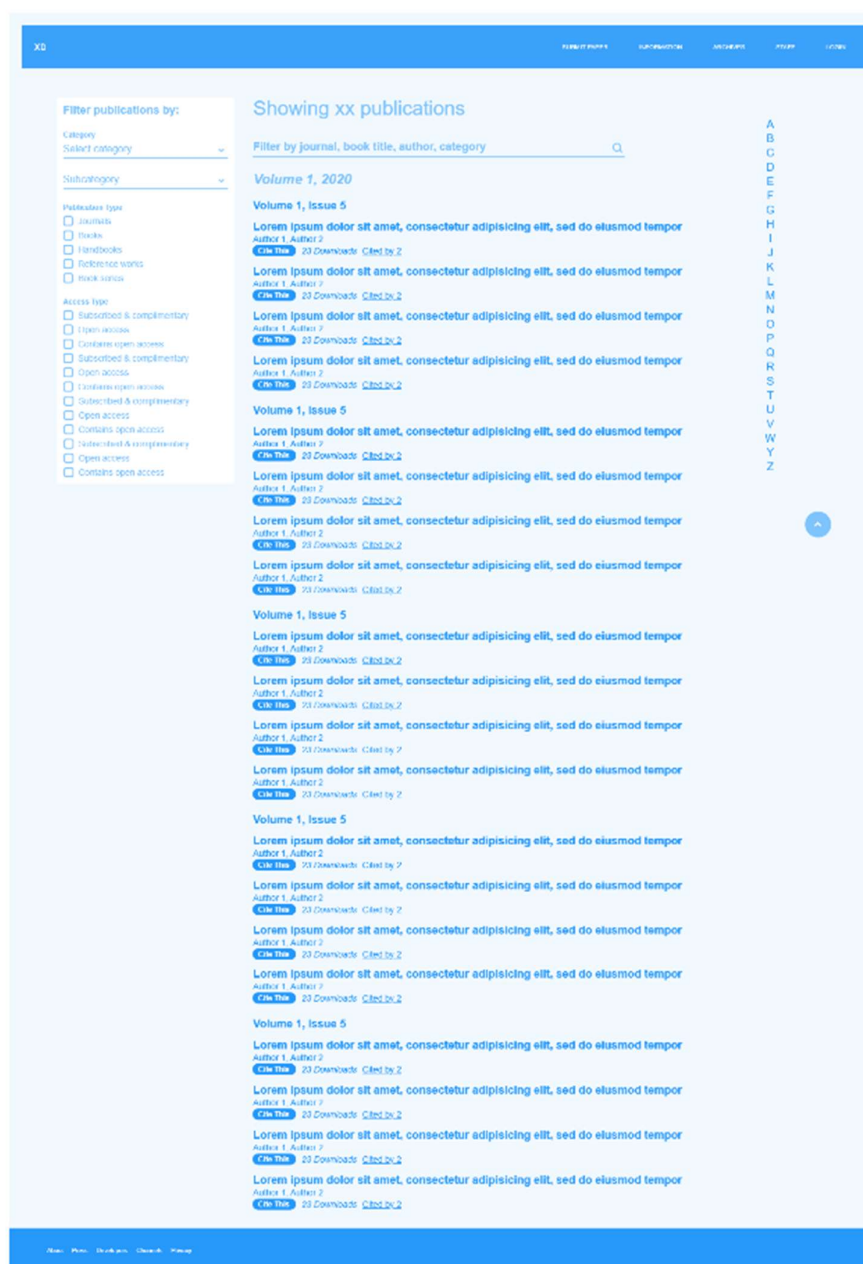


Figure 4.3 Archive Page

JESTEC has published several volumes of article journals and uploads them into their website. To add to the archive page, we've included a filter and search function into the list of journal articles as well as a 'Cite' button for the convenience of the visitors who come to the website to read JESTEC's published journal articles. JESTEC publishes their journal articles in volumes and issues; as such, the archive page follows this by grouping and sorting the articles according to their volume and issue number.

4.1.4 Journal Article Reader Page

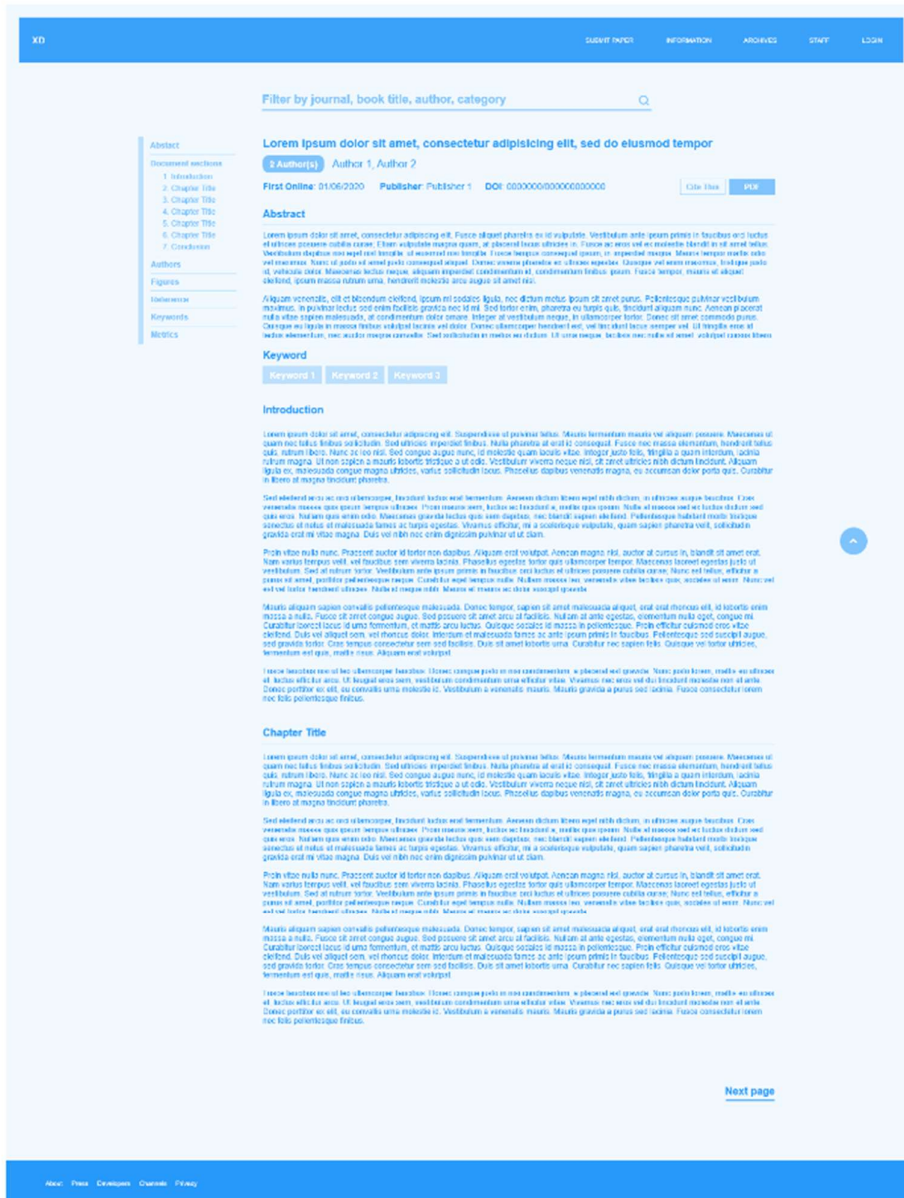


Figure 4.4 Journal Article Reader Page

Visitors are able to read all the published journal articles in the archives, so we’ve made a journal article reader page with the standard of other existing digital libraries such as IEEE. Like many digital libraries, we’ve added the ‘Cite This’ and ‘Download PDF’ functions into the page, as well as a navigation menu on the left for easy navigation. Some of the more important information is displayed on top such as author names and DOI numbers. This all contributes to enriching the visitor’s experience while using our website.

4.1.5 Login Page

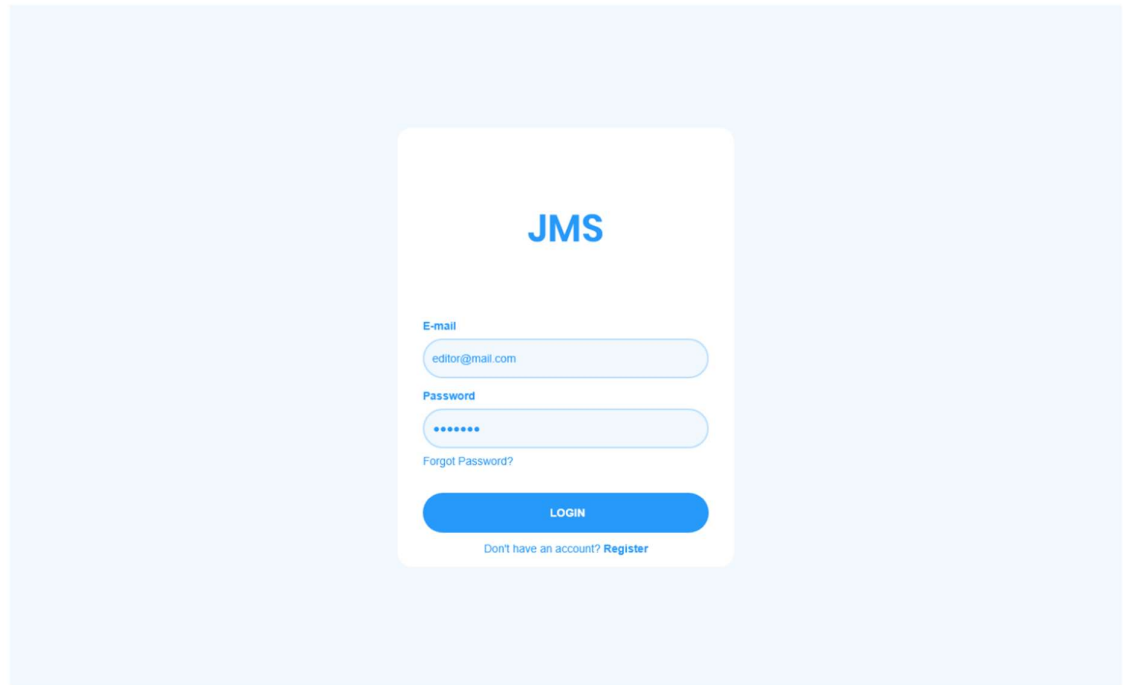
The image shows a login page for a system named JMS. The page has a light blue background. In the center, there is a white rounded rectangle containing the login form. At the top of this rectangle is the 'JMS' logo in blue. Below the logo, there are two input fields: one for 'E-mail' with the placeholder text 'editor@mail.com' and one for 'Password' with placeholder dots. Below the password field is a link for 'Forgot Password?'. At the bottom of the form is a blue 'LOGIN' button. Below the button is a link that says 'Don't have an account? Register'.

Figure 4.5 Login Page

We've decided to keep the login page as simple as possible because users are unlikely to spend much time on this page. However, the team is considering whether to include a Google sign-in function into the login process for faster and easier signups.

4.1.6 Dashboard (Editor)

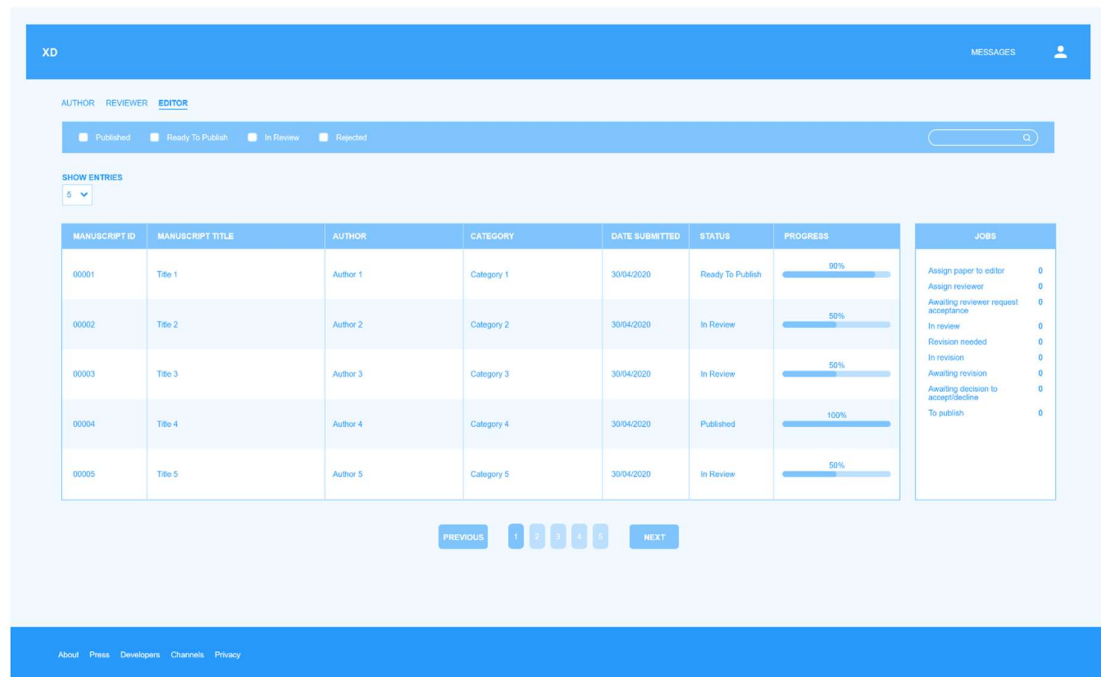


Figure 4.6 Dashboard (Editor)

As an editor, one would need to handle multiple paper submissions at once and would need to look at a lot of information on a daily basis. We've decided to display important information on the dashboard such as the submission's manuscript title, date submitted, status and progress. As mentioned earlier, editors need to deal with a lot of information on daily basis, so having all this information presented in a table to the editor in an organized manner would greatly benefit them.

4.1.7 Dashboard – Search & Filter (Editor)

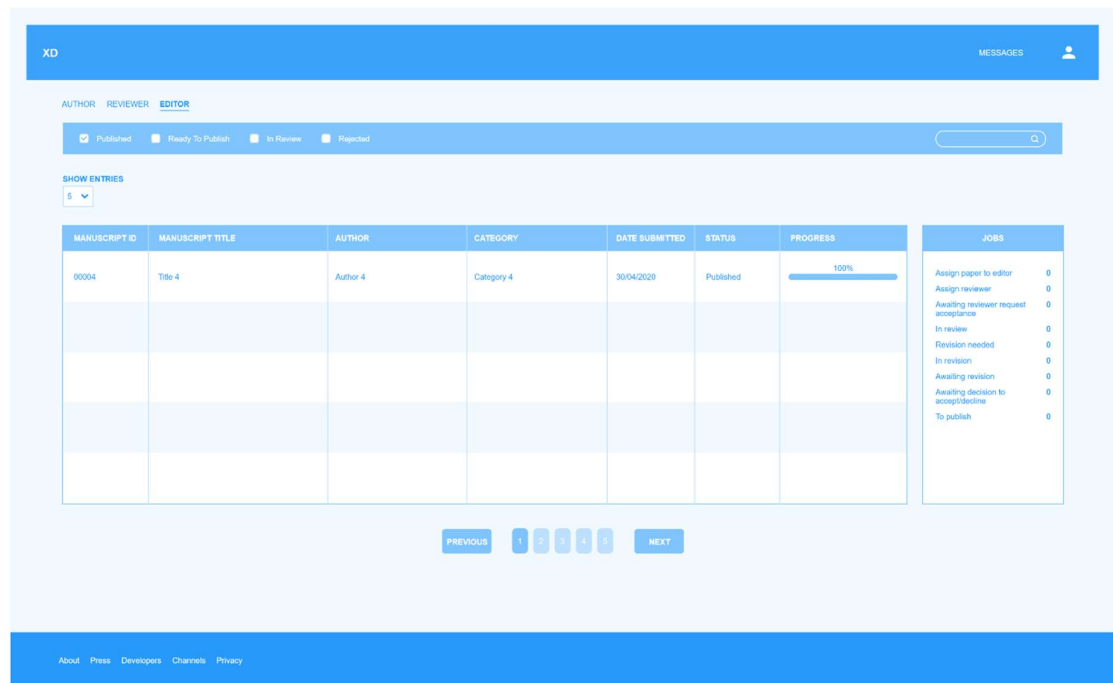


Figure 4.7 Dashboard - Search & Filter (Editor)

To facilitate the editor's tasks, we've implemented a filter and search function that filters out the various submissions that the editor needs to handle by status. For instance, the editor can filter submissions by 'Published' to confirm whether one of the approved submissions are uploaded into the archive or not. Additionally, editors can also just search the submission's manuscript title or ID to quickly look up the information that they need. This would remove the need for editors to look through all of their submissions one by one and in turn, greatly enhance their user experience.

4.1.8 Dashboard – Job Tracker (Editor)

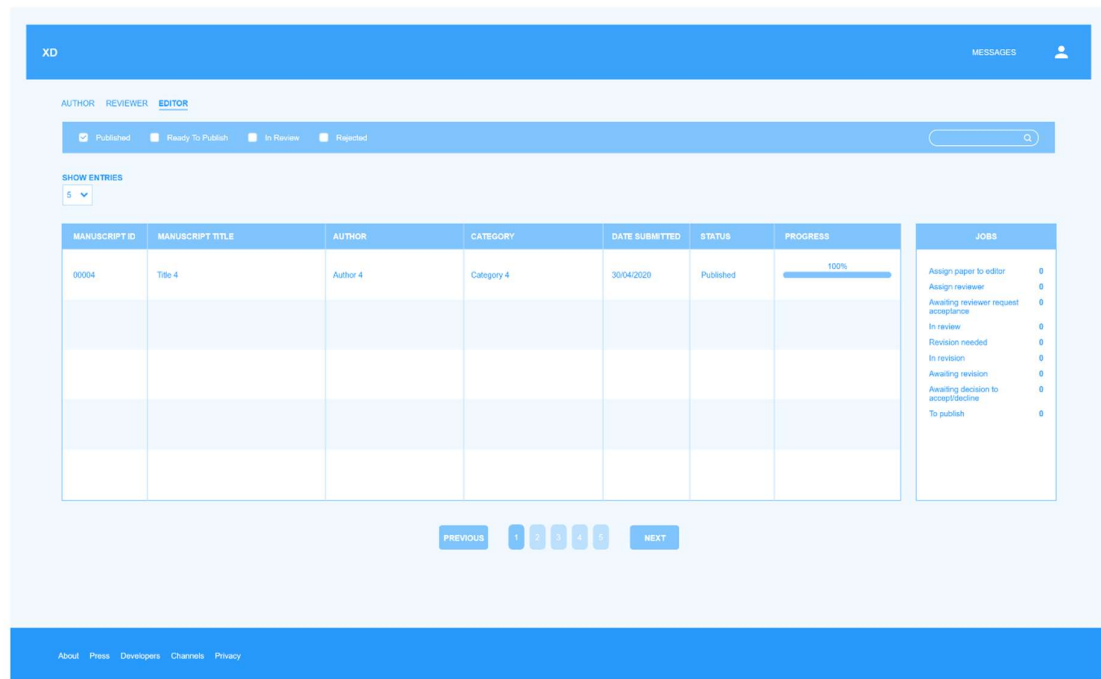


Figure 4.8 Dashboard - Job Tracker (Editor)

An editor may have a large number of submissions at times, so it's easy for editors to get overwhelmed and lose track of the many submissions that they need to handle. To assist them in this matter, we have implemented a job tracker into the editor's dashboard so that they are able to keep track of their jobs and at what stage those jobs are currently in. The editor can click into any of the stages listed on the job tracker and it will redirect them to a page where they can see all the jobs that need to be handled.

4.1.9 Dashboard – Tasks (Editor)

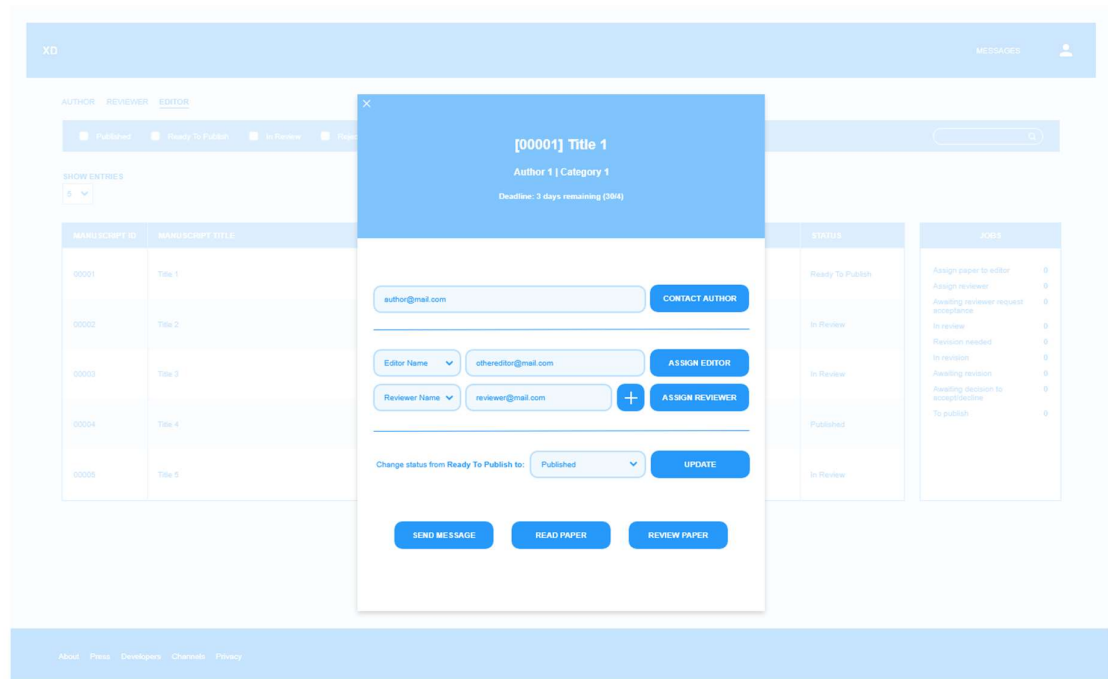


Figure 4.9 Dashboard - Tasks (Editor)

Editors are required to perform various tasks on each of the submissions that they are assigned to, such as:

- assigning associate editors (if they are the chief editor)
- assigning reviewers
- changing the status of the manuscripts

Before, editors would need to perform each task individually via email, which could be troublesome and inconvenient. To circumvent this, we have implemented a pop-up window where editors can perform their tasks all at one place so that the editors can handle the submissions in a more organised manner.

4.1.10 Dashboard (Reviewer)

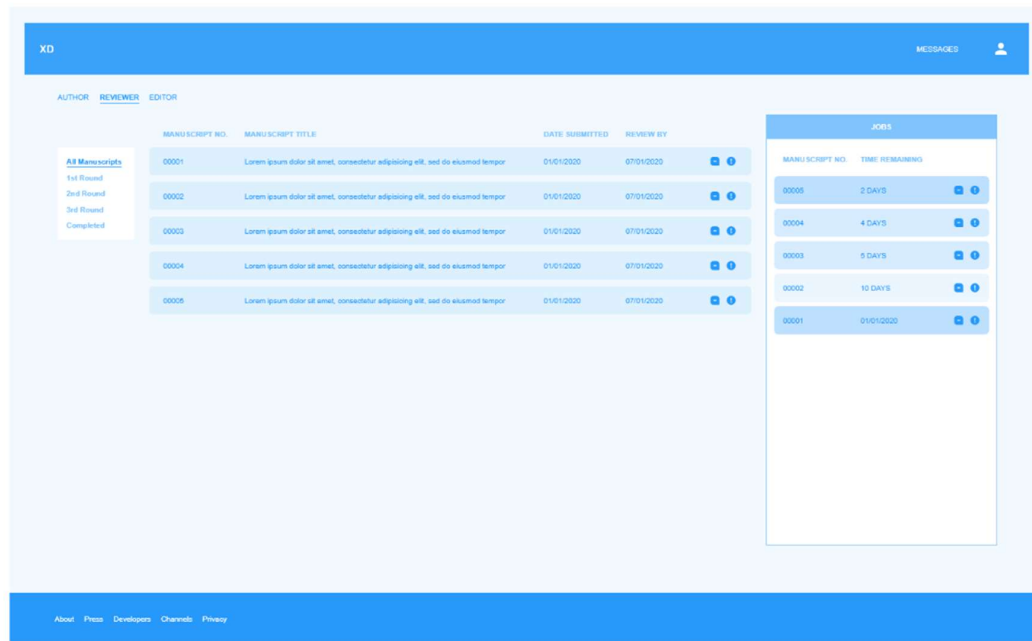


Figure 4.10 Dashboard (Reviewer)

Similar to editors, reviewers also need to review multiple submissions at a given time and would likely lose track of the reviews they need to make due to the process being handled over email. We have implemented a simpler dashboard for reviewers as they do not need to keep track of as many information as editors do; they would only need to know the title of the manuscripts submitted and some important dates.

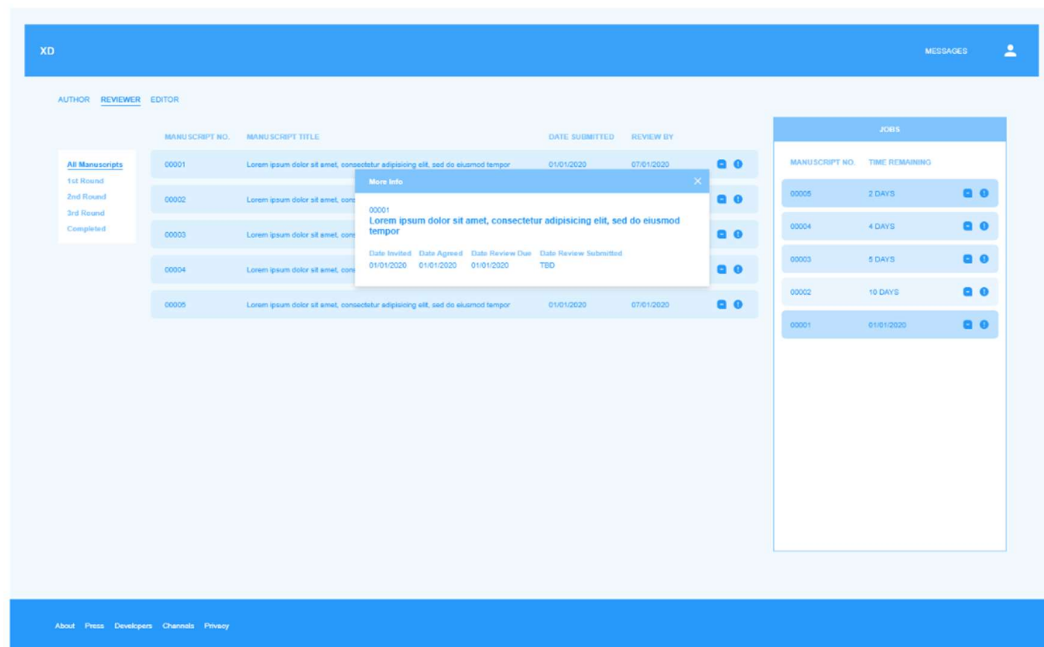
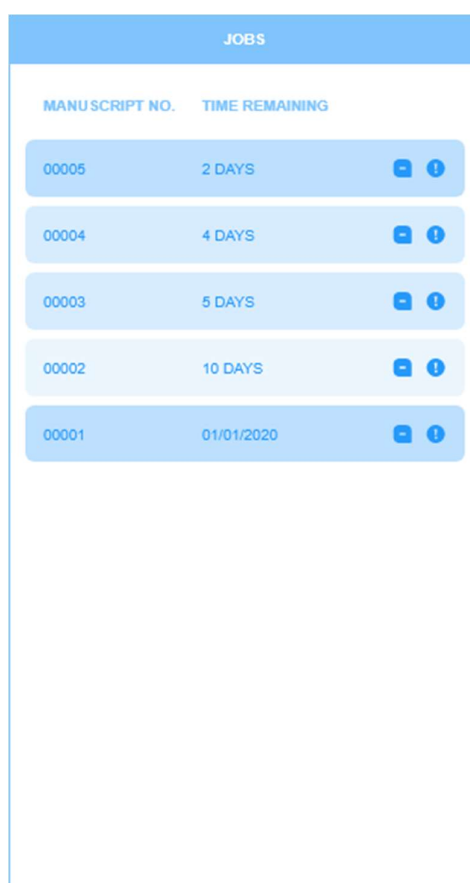


Figure 4.11 Dashboard - More Info (Reviewer)

However, we opted to include a ‘More Info’ button into the dashboard just in case the reviewers would need to retrieve some other information that is not displayed in the dashboard. The reason why we’ve implemented this feature is to reduce visual clutter in the reviewer’s dashboard and avoid distracting the reviewer, making them overwhelmed.

4.1.11 Dashboard – Job Tracker (Reviewer)













JOBS		
MANUSCRIPT NO.	TIME REMAINING	
00005	2 DAYS	 
00004	4 DAYS	 
00003	5 DAYS	 
00002	10 DAYS	 
00001	01/01/2020	 

Figure 4.12 Job Tracker (Reviewer)

As a reviewer, their only task within the process is to review the manuscripts assigned to them by the editors, although they would sometimes be needed to review the manuscript a second or even a third time. To aid reviewers in performing their tasks, we've implemented a job tracker for reviewers as well. The difference between the editor's and the reviewer's job tracker is editors have many different types of jobs whereas reviewers only have one: to review submissions.

As such, we've opted to make the reviewer's job tracker as simple as possible with only 2 columns of information displayed (manuscript number and time remaining) as well as colour coding the jobs according to the status of the submission (darker shade is for 1st round of review whereas lightest shade is for 3rd round of review). The jobs are sorted by ascending time remaining so reviewers can keep track of which submissions are due soon.

4.1.12 Manuscript Reader & Review Form

XD

MESSAGES

Document sections

1. Introduction

2. Chapter Title

3. Chapter Title

4. Chapter Title

5. Chapter Title

6. Chapter Title

7. Conclusion

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce aliquet pharetra ex id vulputate. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia curae. Etiam vulputate magna quam, at placerat lacus ultrices in. Fusce ac eros vel ex molestie blandit in sit amet tellus. Vestibulum dapibus nisi eget nisi fringilla, ut euismod nisi fringilla. Fusce tempus consequat ipsum, in imperdiet magna. Mauris tempor mattis odio vel maximus. Nunc ut justo sit amet justo consequat aliquet. Donec viverra pharetra ex ultrices egestas. Cuiusque vel enim maximus, tristique justo id, vehicula dolor. Maecenas lectus neque, aliquam imperdiet condimentum id, condimentum finibus ipsum. Fusce tempor, mauris et aliquet eleifend, ipsum massa rutrum urna, hendrerit molestie arcu augue sit amet nisi.

Introduction

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse ut pulvinar tellus. Mauris fermentum mauris vel aliquam posuere. Maecenas ut quam nec tellus finibus sollicitudin. Sed ultrices imperdiet finibus. Nulla pharetra at erat id consequat. Fusce nec massa elementum, hendrerit tellus quis, rutrum libero. Nunc ac leo nisi. Sed congue augue nunc, id molestie quam iaculis vitae. Integer justo felis, fringilla a quam interdum, lacinia rutrum magna. Ut non sapien a mauris lobortis tristique a ut odio. Vestibulum viverra neque nisi, sit amet ultrices nibh dictum tincidunt. Aliquam ligula ex, malesuada congue magna ultrices, varius sollicitudin lacus. Phasellus dapibus venenatis magna, eu accumsan dolor porta quis. Curabitur in libero at magna tincidunt pharetra.

Sed eleifend arcu ac orci ullamcorper, tincidunt luctus erat fermentum. Aenean dictum libero eget nibh dictum, in ultrices augue faucibus. Cras venenatis massa quis ipsum tempus ultrices. Proin mauris sem, luctus ac tincidunt a, mollis quis ipsum. Nulla at massa sed ex luctus dictum sed quis eros. Nullam quis enim odio. Maecenas gravida lectus quis sem dapibus, nec blandit sapien eleifend. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vivamus efficitur, mi a scelerisque vulputate, quam sapien pharetra velit, sollicitudin gravida erat mi vitae magna. Duis vel nibh nec enim dignissim pulvinar ut ut diam.

Proin vitae nulla nunc. Praesent auctor id tortor non dapibus. Aliquam erat volutpat. Aenean magna nisi, auctor at cursus in, blandit sit amet erat. Nam varius tempus velit, vel faucibus sem viverra lacinia. Phasellus egestas tortor quis ullamcorper tempor. Maecenas laoreet egestas justo ut vestibulum. Sed at rutrum tortor. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia curae. Nunc est tellus, efficitur a punis sit amet, porttitor pellentesque neque. Curabitur eget tempus nulla. Nullam massa leo, venenatis vitae facilisis quis, sodales ut enim. Nunc vel est vel tortor hendrerit ultrices. Nulla id neque nibh. Mauris et mauris ac dolor suscipit gravida.

Chapter Title

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse ut pulvinar tellus. Mauris fermentum mauris vel aliquam posuere. Maecenas ut quam nec tellus finibus sollicitudin. Sed ultrices imperdiet finibus. Nulla pharetra at erat id consequat. Fusce nec massa elementum, hendrerit tellus quis, rutrum libero. Nunc ac leo nisi. Sed congue augue nunc, id molestie quam iaculis vitae. Integer justo felis, fringilla a quam interdum, lacinia rutrum magna. Ut non sapien a mauris lobortis tristique a ut odio. Vestibulum viverra neque nisi, sit amet ultrices nibh dictum tincidunt. Aliquam ligula ex, malesuada congue magna ultrices, varius sollicitudin lacus. Phasellus dapibus venenatis magna, eu accumsan dolor porta quis. Curabitur in libero at magna tincidunt pharetra.

Sed eleifend arcu ac orci ullamcorper, tincidunt luctus erat fermentum. Aenean dictum libero eget nibh dictum, in ultrices augue faucibus. Cras venenatis massa quis ipsum tempus ultrices. Proin mauris sem, luctus ac tincidunt a, mollis quis ipsum. Nulla at massa sed ex luctus dictum sed quis eros. Nullam quis enim odio. Maecenas gravida lectus quis sem dapibus, nec blandit sapien eleifend. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vivamus efficitur, mi a scelerisque vulputate, quam sapien pharetra velit, sollicitudin gravida erat mi vitae magna. Duis vel nibh nec enim dignissim pulvinar ut ut diam.

Proin vitae nulla nunc. Praesent auctor id tortor non dapibus. Aliquam erat volutpat. Aenean magna nisi, auctor at cursus in, blandit sit amet erat. Nam varius tempus velit, vel faucibus sem viverra lacinia. Phasellus egestas tortor quis ullamcorper tempor. Maecenas laoreet egestas justo ut vestibulum. Sed at rutrum tortor. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia curae. Nunc est tellus, efficitur a punis sit amet, porttitor pellentesque neque. Curabitur eget tempus nulla. Nullam massa leo, venenatis vitae facilisis quis, sodales ut enim. Nunc vel est vel tortor hendrerit ultrices. Nulla id neque nibh. Mauris et mauris ac dolor suscipit gravida.

REVIEW FORM

About

Press

Developers

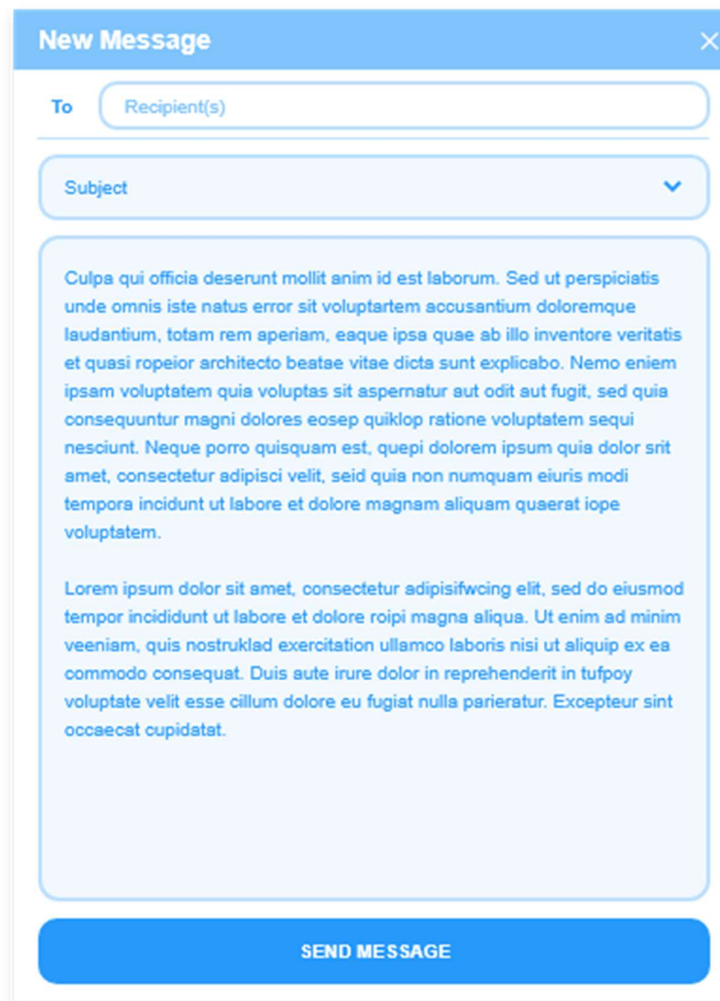
Channels

Privacy

Figure 4.13 Manuscript Reader & Review Form

As a reviewer, in order to make a review, they would obviously need to read through the manuscript and document their comments into a review form. Currently, this process is done by downloading the PDF file of the manuscript and filling in a review form provided by the editor. This process is not necessarily inefficient, and some reviewers may prefer this method than others. However, we will implement a tool for the reviewers so that they are able to read the manuscript and fill in the review form all in one page, in hopes that this would benefit them if they choose to utilize it.

4.1.13 Sending Messages



New Message ✕

To Recipient(s)

Subject ▾

Culpa qui officia deserunt mollit anim id est laborum. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi ropeior architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eoset quiklop ratione voluptatem sequi nesciunt. Neque porro quisquam est, quepi dolorem ipsum quia dolor srit amet, consectetur adipisci velit, seid quia non numquam eiuris modi tempora incidunt ut labore et dolore magnam aliquam quaerat iope voluptatem.

Lorem ipsum dolor sit amet, consectetur adipisifwcing elit, sed do eiusmod tempor incidunt ut labore et dolore roipi magna aliqua. Ut enim ad minim veeniam, quis nostruklad exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in tufpoy voluptate velit esse cillum dolore eu fugiat nulla parieratur. Excepteur sint occaecat cupidatat.

SEND MESSAGE

Figure 4.14 Sending Messages

All users (authors, reviewers and editors) involved in this system have the capability to send messages to each other; pre-determined messages which vary for each of the different types of users such as accepting a review request (reviewers), informing authors of the results of their submission (editors), and many more. Currently, this process is mainly handled manually over email which involves relying on third-party applications (such as Gmail, etc.) as a means of communication. To further streamline the peer-review process, we've decided to integrate a means of communication within the system. By doing so, we are able to automatically fill up certain information involving the submission in the message template as well as removing the need to alternate between the system and third-party applications.

4.1.14 Messages Page

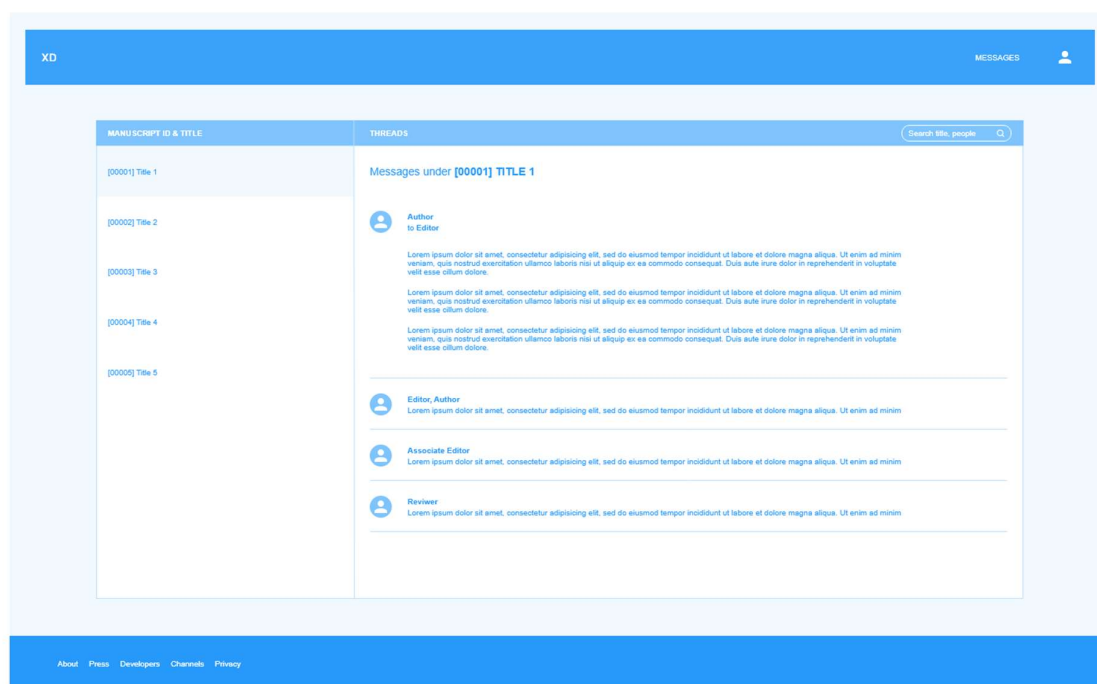


Figure 4.15 Messages Page

It is very likely that all users in the system would be involved in multiple submission at once: authors could submit multiple manuscripts; reviewers are likely to review several manuscripts at once; and editors would definitely handle more than one submission at a time. Due to this, it can be very confusing to sort through the messages involving the many submissions. To facilitate this issue, we've implemented an inbox system that compiles the various messages the users receive into threads belonging to the respective submission. This makes it easier to keep track of the messages received on any particular submission, thus reducing the possibility of miscommunication between users.

4.1.15 Profile Page

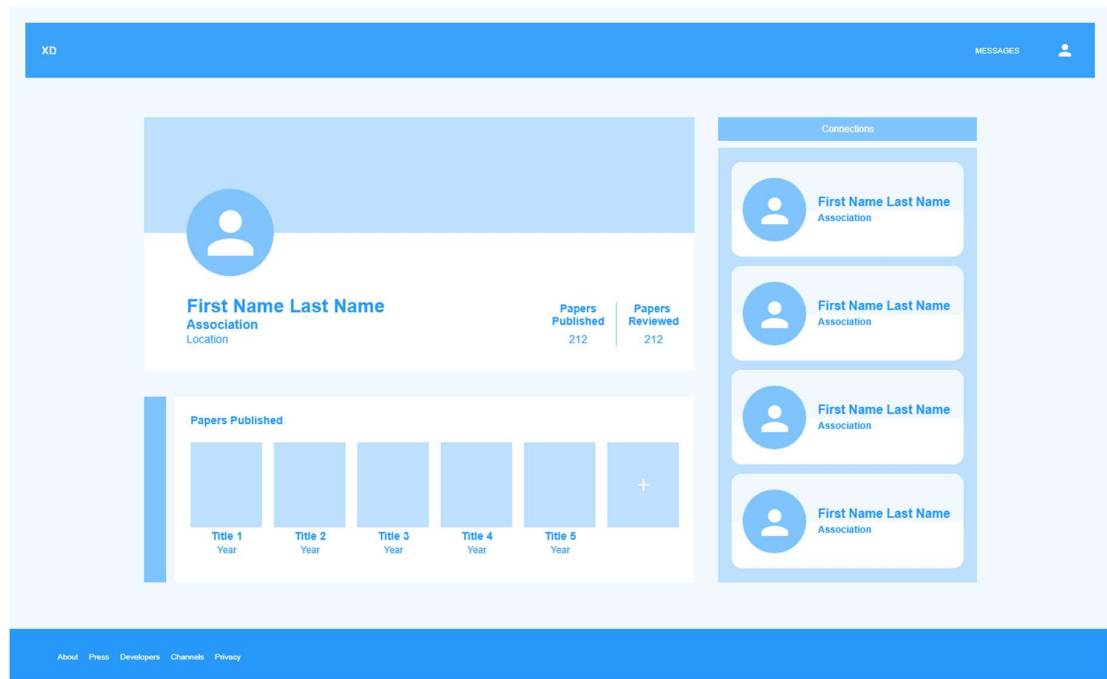


Figure 4.16 Profile Page

As all users are required to register an account to access the main features of the website, we have implemented a profile page where users can display their information. In this page, the user's stats on how many papers that they have published (editor), reviewed (reviewer) or submissions that were approved (author) are displayed as well as the papers that they have submitted/published/reviewed. The information displayed on this page serves as a way for users to show their contribution in JESTEC and could be used as a portfolio.

4.2 Test plan design

4.2.1 Introduction

This test plan is designed to describe the scope of the testing that will be conducted during the testing phase in order to ensure that the system that we have developed works according to the specifications. In this section, we will be discussing our approach for testing our system, JMS for JESTEC.

4.2.2 Objective

The objective of this test plan is to define:

- the scope of testing
- the testing strategy that will be used during testing
- the testing environment needed during testing

The purpose of our system is to facilitate the end-users in their daily tasks as authors, editors and reviewers. Thus, the testing will need to ensure that the system is convenient to use and works as it is intended.

The mode of communication between the testing team will be primarily via online communication such as video conferences, instant messaging, etc. Besides that, the team will conduct meetings in person when the opportunity arises.

The table below describes the testing team members.

Table 4.1 Testing Team Members

Name	Roles
Prof. Dr. AbdulKareem Sh. Mahdi Al-Obaidi	Project Sponsor
Dr. Noor Zaman Jhanjhi	Project Supervisor
Fauzi Anuar Lee	UI/UX Designer, System Analyst, Programmer, Tester
Jacky Yap	Data Engineer, System Analyst, Programmer, Tester
Muhammad Yazid bin Zaini	Software Security Analyst, Security Programmer, System Analyst, Programmer, Tester

4.2.3 Scope

The scope of the testing will be:

- All functions in the system
- User interface of the system
- Integration between cloud service provider and database

4.2.4 Testing Strategy

4.2.4.1 Unit Testing

Unit testing is a type of testing that tests every individual function in the system to determine whether the function is working as specified and will be carried out on all of the individual functions.

4.2.4.2 Integration Testing

Integration testing is a type of testing that tests the interaction between two or more functions to determine whether the functions are integrated together properly and working as intended. This testing will primarily be used for the integration between the database system and the cloud service provider.

4.2.4.3 System Testing

System testing is a type of testing that tests the system as a whole and whether the system fulfils all requirements and works according to the specifications. This testing will be done during the final stages of development.

4.2.4.4 Performance Testing

Performance testing is a type of testing that tests the system's capabilities and upper limits. This testing will primarily use stress testing to evaluate the behaviour of our system when we surpass the limits of its anticipated workload.

4.2.4.5 Acceptance Testing

Acceptance testing is a type of testing that tests whether the system adheres to business requirements and is ready for deployment. This testing will be done by end-users to evaluate their satisfaction on the product.

4.2.5 Testing Environment

The testing environment for our system will be on any web browser (e.g. Google Chrome) installed in a Windows 10 system with at least 4gb of RAM. The table below lists down the members that will be performing the tests.

Table 4.2 Testing Assignment

Task	Contributors
Approval of Test Cases	Fauzi
Unit Testing	Fauzi, Jacky, Yazid
Integration Testing	Fauzi, Jacky, Yazid
System Testing	Fauzi, Jacky, Yazid
Performance Testing	Fauzi, Jacky, Yazid
Acceptance Testing	Fauzi, Jacky, Yazid

References

- Alonso-Virgós, L., Thomaschewski, J., Espada, J. P., & Crespo, R. G. (2020). Test usability guidelines and follow conventions. Useful recommendations from web developers. *Computer Standards and Interfaces*, 70(November 2019), 103423. <https://doi.org/10.1016/j.csi.2020.103423>
- Bootstrap (n.d.). Getting started. Retrieved June 17, 2020, from <https://getbootstrap.com/docs/4.5/getting-started/introduction/>
- Brown, K (2019). What Is GitHub, and What Is It Used For? Retrieved June 17, 2020, from <https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/>
- Clark, J. W., & Stephane, A. L. (2018). Affordable Eye Tracking for Informed Web Design. In A. Marcus & W. Wang (Eds.), *Design, User Experience, and Usability: Theory and Practice* (pp. 346–355). Springer International Publishing.
- Faisal, C. M. N., Gonzalez-Rodriguez, M., Fernandez-Lanvin, D., & de Andres-Suarez, J. (2017). Web Design Attributes in Building User Trust, Satisfaction, and Loyalty for a High Uncertainty Avoidance Culture. *IEEE Transactions on Human-Machine Systems*, 47(6), 847–859. <https://doi.org/10.1109/THMS.2016.2620901>
- Google Analytics (2020). Bounce Rate. Retrieved May 29, 2020, from <https://support.google.com/analytics/answer/1009409?hl=en>
- Hofstede, G. (n.d.). *Culture's consequences : comparing values, behaviors, institutions, and organizations across nations*. Second edition. Thousand Oaks, Calif. : Sage Publications, [2001] ©2001. <https://search.library.wisc.edu/catalog/9910023165902121>
- Lindgaard, G., Fernandes, G., Dudek, C., & Brown, J. (2006). Attention web designers: You have 50 milliseconds to make a good first impression! *Behaviour & Information Technology*, 25(2), 115–126. <https://doi.org/10.1080/01449290500330448>
- Lodhi, A. (2010). Usability Heuristics as an assessment parameter: For performing Usability Testing. *2010 2nd International Conference on Software Technology and Engineering*, 2, V2-256-V2-259. <https://doi.org/10.1109/ICSTE.2010.5608809>
- Mozilla (2020). HTML: Hypertext Markup Language. Retrieved June 13, 2020, from <https://developer.mozilla.org/en-US/docs/Web/HTML>
- Mozilla (2019). HTML5. Retrieved June 13, 2020, from <https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5>
- Mozilla (2020). CSS: Cascading Style Sheets. Retrieved June 13, 2020, from <https://developer.mozilla.org/en-US/docs/Web/CSS>

- Nielsen, J. (1994). Enhancing the Explanatory Power of Usability Heuristics. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 152–158. <https://doi.org/10.1145/191666.191729>
- Oyibo, K., & Vassileva, J. (2017). What Drives Perceived Usability in Mobile Web Design: Classical or Expressive Aesthetics? In A. Marcus & W. Wang (Eds.), Design, User Experience, and Usability: Theory, Methodology, and Management (pp. 445–462). Springer International Publishing.
- Pang, X., Cao, Y., Lau, R. W. H., & Chan, A. B. (2016). Directing User Attention via Visual Flow on Web Designs. ACM Trans. Graph., 35(6). <https://doi.org/10.1145/2980179.2982422>
- Rails (n.d.). Getting Started with Rails. Retrieved June 13, 2020, from https://guides.rubyonrails.org/getting_started.html
- Ruby (n.d.). About Ruby. Retrieved June 13, 2020, from <https://www.ruby-lang.org/en/about/>
- Start Bootstrap (n.d.). Main Page. Retrieved June 17, 2020, from <https://getbootstrap.com/docs/4.5/getting-started/introduction/>