Grenze International Journal of Engineering and Technology, January Issue

All-in-One (AIO): A Software Engineering Approach to Unified Bookmark Management and Smart Retrieval Platform

Norah H. Alanzi[1], Wujud K. Aljarah[2], Rawan D. Alaqeel[3], Enas E. ElSharawy[4] and Lojain H. Alibrahim[5]
[1-5]Computer Science Department, College of Science and Humanities, Imam Abdulrahman Bin Faisal University, P.O.Box 31961, Jubail, Saudi Arabia
Email: 2220000572@iau.edu.sa, {2220001292,2210003568, eelsharawy, 2220001991}@iau.edu.sa

***Abstract*— The rapid increase in social networks has significantly transformed the creation, sharing and consumption of digital content. Although these platforms give users a wide range of communication and sharing of resources, they still lack efficient methods for systematic organization and easy retrieval of saved content. This shortcoming has led to a huge demand for well-coordinated systems that can help in the problems of too much information and disjointed digital resources. This paper presents All-in-One (AIO), which is a comprehensive bookmark management system designed to integrate, categorize, and manage digital media from various social media platforms. In this research, the Waterfall Model has been used in the software development process, which structures the development process as sequential stages in an organized process that requires completing all activities and the planning stage before continuing the development process. The methodology involves four phases: Software Project Management Planning (SPMP), Software Requirements Specification (SRS), Software Design Specification (SDS) and Software Test Planning (STP). The AIO platform addresses information overload by providing a platform that allows the organization of bookmarks in multiple social media sources, allowing users to efficiently save, categorize, and search for saved media using both automatic and manual classification approaches. The research enhances the digital content management, by providing a scalable platform that improves the productivity and user experience in bookmark organization.**

***Index Terms*— Bookmark Management, Digital Content, Social Media, Waterfall Model.**

I. INTRODUCTION

Technology has become a fundamental pillar of modern life given that it influences how people tend to work, communicate, and manage all of their daily activities. Society's aspects all include its integration, streamlining processes, improving access, and connecting everyone more [1]. Technology allows individuals as well as organizations to plan, prioritize, and achieve goals more effectively [2] when paired with strong organizational practices. Digital tools with management strategies improve productivity and encourage innovation [3]. Research shows these systems can adapt to the rapid pace of change. In today's digital world, the rapid expansion of social media platforms has led the individuals to interact and engage with an overwhelming amount of digital content. This continuous engagement often leads the users to save videos, posts, and other digital content for future use.

Grenze ID: 01.GIJET.12.1.23

However, managing saved information is difficult, and users frequently create bookmarks but rarely return to them because of poor organization, and the search mechanism is flawed [4]. This issue develops worse with information overload, caused by the constant increase in digital content, leading to a decrease in decision-making quality and productivity [5].
In this paper, we propose a system design that aims to help social media users in managing their scattered bookmarks from several social media platforms within a single reliable platform called All-in-One (AIO). This platform integrates personal bookmark management with social interaction, which allows users not only to save bookmarks but also to publish and share bookmarks with other users.

II. Problem Statement

By the year 2025, the number of social media users has reached 5.41 billion worldwide, which represents 65.7% of the world population [6]. This indicates the significant increase in reliance on digital platforms for communication, content sharing, and learning. Also, with this enormous growth in the digital content field, the individuals face a growing challenge in organizing the content, links, and digital resources they save, especially with the scattered content across different platforms and the shortcomings of current tools that lack integration, intelligent categorization, and effective search. This shortage leads to the loss of important information and reduces the use of the digital resources, which highlights the need to develop an integrated platform such AIO. This platform will provide a unified environment that facilitates saving, categorizing, searching, and retrieving links in a more flexible and efficient way.

III. Research Aim and Objectives

This platform aims to maximize the utilization of digital content that individuals engage with daily. It provides many features to the users that will facilitate the retrieval process. The main feature is the ability to save the bookmark content from various resources in one integrated platform with many saving format options. In addition, it provides the user with intelligent bookmark categorization into main and subcategories with the ability to customize the categories and the bookmark information and link it with different apps such as Google Maps for efficient access.
The users can also add notes to the bookmark information and filter the content based on keywords for faster retrieval. Moreover, users can make their own page visible to the other users to show their favourite savings and interact with others. The AIO platform aims to improve productivity and enhance the user experience by integrating advanced organization and search functionalities into one seamless application.

- Objective 1: Centralize bookmark management.
- Objective 2: Implement smart and flexible classification.
- Objective 3: Provide quick and advanced search and filtering.
- Objective 4: Enable notes and descriptions on bookmarks.
- Objective 5: Improve user experience with personalization and sharing.

IV. Related Work

In [5], the authors focused on the long-standing problem of information overload becoming increasingly pressing in the digital era. The study creates cognitive strain, reduced user productivity, and weakened effective decision-making due to the enormous amount of data available online. Because of how they drew on both historical and also modern perspectives, the authors highlighted the fact that digital technologies and those social media platforms have exacerbated information overload, and this is not a new phenomenon. Their analysis pointed to a few strategies for addressing the problem that included filtering mechanisms as well as improved categorization and prioritization of relevant content with better design of information systems. These findings emphasize the importance of creating tools that enable users to efficiently manage large datasets. Reducing the psychological burden from excessive information exposure is also important.
In [7], the authors focused on systematically reviewing the factors that affect the people's intentions regarding knowledge sharing on social media platforms by using the ROSES protocol. The study explores high-quality studies that are mostly quantitative, with the Zhihu platform being the most frequently examined. The researchers identified four groups of factors, which are the psychological, technological, environmental, and social, with trust and attitude being the most common influences. The results revealed that trust and a positive attitude were the most influential, in addition to performance expectations and a sense of platform usefulness, which encouraged individuals to participate more. The authors concluded that behavioral theories explain the

individual desire to share knowledge better than mass communication frameworks. They also offered recommendations for enhancing interaction and encouraging knowledge sharing through effective strategies on social media and emphasized the importance of fostering positive social interactions and a community-oriented environment.

In [8], the authors focused on studying the problem of website refinding and the rare use of bookmarks compared to other access methods. They designed a quantitative controlled study with 50 students using the Elicited Personal Information Retrieval (EPIR) technique to recover different types of websites and measuring the success rates, time, and used methods for retrieval.

The study showed that the bookmarks were faster and never failed when used; however, only 16% were actually retrieved using bookmarks, and they relied heavily on visibility in the browser bar, while the remaining 84% were retrieved using other methods. The study indicated that the participants preferred orienteering strategies where the target is approached by a series of gradual steps like searching and browsing, and often bookmarked sites that were already easy to find. The authors concluded that although bookmarks can be efficient, they are not effectively used, and suggested design improvements such as highlight reels and ranking bookmarked sites higher in search results to better utilize them.

In [9], the authors focused on addressing the problem of automating the feature selection in deep recommender systems. They introduced AutoField, which is an AutoML-based framework that was designed to automatically identify the most predictive feature fields by a differentiable controller network. The proposed model performance was evaluated on two benchmark datasets, where the results have shown that AutoField has improved the recommendation accuracy and reduced the inference time by choosing informative fields. Additionally, the authors have discussed that the selected features could be transferable to various recommendation models to enhance the performance and reduce computational cost.

In [10], the authors focused on automatic hashtag recommendations of social media content through a multimodal learning approach. They proposed LXMERT4Hashtag, which is a cross-modality representation learning transfer model that aims at enhancing the hashtag recommendation task by integrating image features with textual captions. The model was assessed by comparing the proposed approach with existing state-of-the-art baseline methods, and the results indicated that LXMERT4Hashtag had achieved higher performance across all the benchmark metrics.

In [11], the authors introduced OrgBox, a knowledge representation tool designed to assist users during complex search operations. The tool was integrated into a custom-designed search system that allowed participants to save passages, organize them into thematic “boxes,” create hierarchies, and add notes. According to the study, OrgBox allowed users to externalize and rearrange knowledge while searching, in contrast to a baseline bookmark tool that only saved text fragments. Findings from a lab study with 32 participants indicate that OrgBox significantly enhanced cognitive and metacognitive activities, including information organization, making connections, planning, monitoring progress, and revising strategies. The results further suggest that OrgBox promoted deeper learning, more effective outline creation, and improved task organization.

In [12], the authors focused on the difficulties of managing complex online web resources in online learning and academic research.

They developed Bewilder, a system that captures metadata from web links and presents it in a visual interface. The tool was developed for researchers and students who often find themselves working with a large amount of online resources. By helping users group, annotate and navigate collections of links better, Bewilder reduced the time and effort in finding previously saved resources.

The findings from user evaluations indicated substantial improvement in resource retrieval and productivity as a whole. Overall, the study illustrated the utility of visual- and metadata -based organization to counter some of challenges presented by traditional text-based bookmarks.

In [13], the authors introduced Scrapbook, an application designed for digital resource curation across multiple applications through the use of screenshot-based bookmarks. The study explains how the system captures visible windows on a user’s screen, extracts metadata such as URLs, file paths, and application names, and organizes them into a visual bookmark. The results show that Scrapbook helps users to restore cross-application working environments more efficiently than traditional text-based bookmarks that are restricted to a single program. Results from a one-week field study with 13 participants indicate that screenshot-based bookmarks improved resource retrieval, facilitated task resumption, and provided richer visual and textual cues that enhanced productivity.

In [14], the developers introduced Linkapp, a mobile link management application that enables users to save and organize links across different platforms. The platform allows users to allocate a separate folder for each supported application, particularly popular social media apps, and add links without storing the actual content of

those links. Unlike traditional bookmarking tools, Link App provides additional features such as tracking link performance through URL details, searching by link or name, and sharing saved links in QR code format. The program also allows for folder classification; however, it is restricted to a pre-made list and does not allow for custom folder names. Overall, Link App offers a structured approach to link management, enabling users to organize, monitor, and share their online resources more effectively.

In [15], the developers introduced Pinterest, a visual bookmarking platform that replaces conventional text-based methods by allowing users to "pin" and organize content into thematic "boards." With millions of active users worldwide, the platform has become popular for its intuitive design and ease of sharing visual ideas across communities. Studies on its adoption show that visual curation contributes to improved recommendations, enhanced discovery, and more engaging user experiences.

In [16], the developers presented Hammel, a mobile file management platform that provides users with an integrated platform for downloading files on mobile devices. The Hammel application allows the user to import files from various platforms or import directly from the saved files on the device. Furthermore, the application supports importing multiple file formats, such as videos, images, PDFs, and text files. Additionally, the application includes functionalities such as saving external links for later access, scanning barcodes from the uploaded images, and video editing. Moreover, the Hammel platform provides options for password protection and face recognition to enhance the security of sensitive data.

In [17], the developers introduced Raindrop.io, a cloud-based bookmark management platform that allows users to save and organize web content. The platform enables users to collect and share URLs, articles, and images while synchronizing bookmarks across multiple devices. It also supports categorization through collections and tags, assisting faster retrieval and efficient organization.

Additionally, Raindrop.io provides search functionality; however, some advanced search features, such as searching within saved content, are restricted to paid versions. Users can also share bookmarks as folders, which may be private or public.

Nevertheless, Raindrop.io relies primarily on the manual categorization and user input, which can be time-consuming when managing a large number of links. The platform focuses on saving the URLs rather than the actual content, and it does not offer options for saving different content formats.

Table I presents a comparative analysis of bookmark management platforms mentioned in the related work in terms of supported features.

TABLE I. COMPARISON OF BOOKMARK MANAGEMENT PLATFORMS

Feature / Platform	Raindrop.io	LinkApp	Pinterest	Hammel	AIO (PP)
Multi-platform bookmark saving	✓	✗	✓	✓	✓
Custom category	✓	✗	✓	✓	✓
Sub-category support	✗	✗	✗	✓	✓
Multiple bookmark formats	✗	✗	✗	✓	✓
Keyword search	✓	✗	✓	✓	✓
Notes and descriptions on bookmarks	✓	✗	✗	✗	✓
Social sharing	✓	✓	✓	✓	✓
Manual and automatic classification	✗	✗	✓	✗	✓
Unified management across platforms	✓	✗	✗	✓	✓
Primary focus	Cloud bookmark storage	Social communication and content sharing	Visual content discovery	Mobile file management	Unified smart bookmark management

V. METHODOLOGY

The AIO will use the plan-driven or plan-based Incremental Development Process Model. This model focuses on detailed plans throughout making the software. All tasks, responsibilities, timelines, and goals are clearly documented. This approach was chosen because the software requirements are clearly defined and easily understood, removing any mix-ups later. The project has clear tasks to complete, with steps done one after another without mixing, as seen in Fig. 1, in addition to the research methodology which consists of four main stages, as shown in Fig. 2

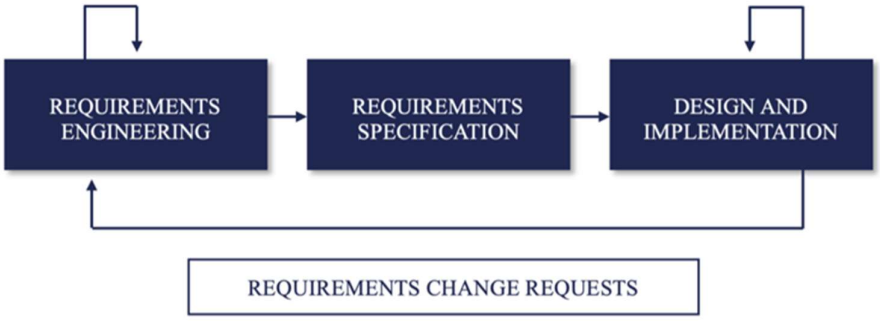

Figure 1. Plan-driven Model Process Phases. Adapted from [18]

Figure 2. Research Methodology

A. Software Project Management Plan

Throughout this section, a comprehensive analysis of the project's deliverables, along with an outline of the schedule and budget, as well as the progression of the plan over time, is also presented. The software development process is thoroughly described in this section, combined with identifying the organizational boundaries between the project and external entities, as well as the internal structure of the project organization. It will also define the roles and responsibilities of the team members in line with their qualifications. This part of the Project Management Plan identifies the project management processes for the project. It also outlines the plans for project start-up, risk management, project work, project tracking, and project close-out. The technical process plans section outlines the strategies for executing the project. It provides a comprehensive explanation of the chosen process model, which defines the relationships between activities and work products. It also specifies methods, tools, and techniques to be used in development and the infrastructure required to support the project. Furthermore, it details the plan for customer acceptance of project deliverables.

Estimates

To manage the project efficiently and effectively, it's essential to assess the initial estimates. The project includes multiple phases that are adjusted and updated as work progresses. Initially, estimating the project cost requires identifying the necessary hardware, software, and resources. Additionally, the project participants will need training on this intensive program. Assume that the project requires integration with a social media platform so that AIO appears in the native share menu; the project will implement a social share extension to provide direct access to the users. The project will require the use of suitable hardware, such as a computer possessing elevated specifications to effectively handle substantial software tools. The development of an AIO application, which includes features like searching for links using keywords and filtering results, will require the tools necessary to program and implement these ideas. This will include MySQL for data collection and storage, Java Database Connectivity (JDBC), CSS, and the Java programming language using the NetBeans development environment.

Requirements Management

To ensure effective requirements management, the project team will be fully informed and always ready to deal with any changes in requirements. To illustrate in Fig. 3, which shows the status of requirements, at the first stage the requirements are presented, and in the next stage the requirements are either accepted or rejected. If they're approved, we will move to the implementation stage. In the implementation stage, if it is decided not to implement it, the requirement will be deleted, but if it is decided to implement it, the team will start preparing, designing, and testing without compromising the original goal. In the final stage, the team must decide whether to approve the requirement for current or future implementation.

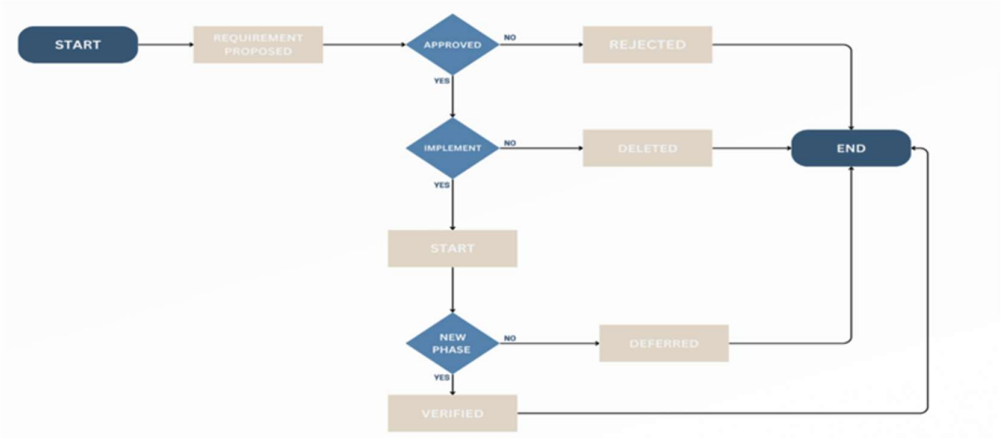

Figure 3. Requirement's Status.

Risk Management Plan

The working team must also consider risk management when analyzing project requirements, taking into account how they affect the project schedule, budget, risk factors, and resources. Risk management is the process of identifying risks, assessing their likely impact, and planning strategies to manage them [18]. It includes creating strategies to reduce or control the risks to ensure operational stability while also achieving the organizational goals. Since the risks can change or become more significant during a project, risk management calls for ongoing observation. The first phase is risk identification, in which project, product, and business risks are clearly identified. The second phase is risk analysis, during which the likelihood and consequences of the identified risks are assessed using preexisting information or by utilizing tools and techniques such as fault tree analysis. The third phase is risk planning, where strategies are developed to avoid or reduce the effects of the risks. The final phase is risk monitoring; as mentioned earlier, risks can change during the project, so the identified risks from the first phase are monitored and tracked throughout the project. Table II outlines the potential risks associated with these processes.

TABLE II. RISK MANAGEMENT

#	Risk	Probability	Effects	Action	Avoidance
1	requirements change	Low	Tolerable	Identify the challenges and impacts of modification with the customer.	Clear requirements and standards should be defined in the planning phase.
2	Limited experience	Moderate	Serious	The project manager should consult with the supervisor.	Before writing the report, the team should review the project, consult the supervisor if unclear, and provide training for missing skills.
3	Failing to meet the predefined schedule	Low	Serious	Adjust the time and date.	Use scheduling tools (e.g., Microsoft Planner) to set reminders.
4	Team members' health condition	Moderate	Serious	Redistribute the rest of the work among the team members.	Keep all members updated so they can cover for each other if needed.
5	Cash flow problems	Low	Catastrophic	Reschedule from the initial stage of the cost and budget.	From the start, decide how much budget can be set aside.
6	Integration challenges	Moderate	Serious	Conduct thorough integration testing.	Design modular architectures.
7	Tools unavailable due to budget limits	High	Tolerable	Use alternative tools with similar quality.	Allocate sufficient additional budget for tools.

B. Software Requirements Specification

In this section, the AIO system software requirements are provided and described in detail. The section defines the system purpose, scope and product perspective, software interfaces, functionality and the specific requirements. The purpose of the SRS document is to simplify the load associated with the development,

validation, and verification operations. Therefore, the development team follows the documented requirements to create an organized basis for the implementation phase [18]. The software is designed with two user roles: users who manage their bookmarks and system administrators who manage the system operations through admin dashboards. In the SRS documentation, we have identified nine core functional requirements, which include user authentication, bookmark management, and system administration capabilities. To demonstrate how the specified functional requirements translate into an actual system interaction, we have conducted a use case diagram as shown in Fig. 4 that illustrates the relationship between the system actors and the functional requirements. After that, sixteen distinct user interactions were analysed in use case tables. Each table provides insights about the use case goal, identify who interacts with the system, preconditions and post-conditions, and triggers. In addition, non-functional requirements are specified such as performance, reliability, availability, security, maintainability and portability. Fig. 5, illustrates the Functional and Non-Functional requirements of AIO.

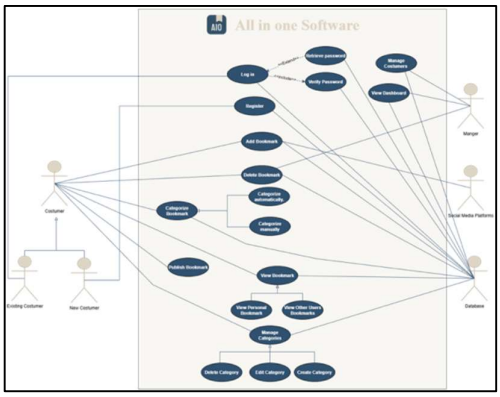

Figure 4. Use Case Diagram

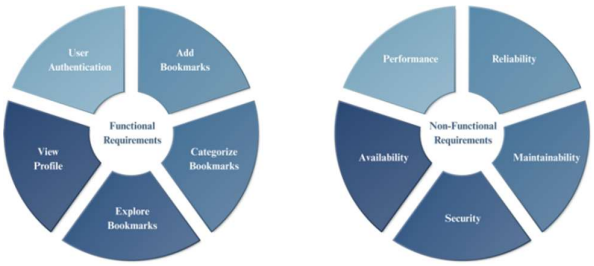

Figure 1.Functional and Non-Functional Requirements

The specifications included a detailed design documentation such as:

- UML Class Diagram: this diagram identifies the system structure which includes five classes where each class represents an entity of the application's functionality.
- Entity Relationship Diagram (ERD): this diagram identifies the logical database requirements and contains four entities (Manager, Customer, Bookmark, Category) and six relationships.
- Sequence Diagrams: this diagram specifies how the actors and objects interact with the system [18].
- Data Flow Diagrams: this diagram identifies how the data moves in the system during a specific use case.

C. Software Design Specifications

This section fully depicts AIO software in detail from many perspectives. It will enclose user interface design, data design, system architecture, and how to align the SRS to the Software Design Specification (SDS). Several models contributed to the system's architecture demonstration. These models show the inner design and data flow of the application and include ERD, Relational Mapping, and Sequence Diagrams. Furthermore, diagrams are used to determine specific requirements.

- System Architecture: The client-server architecture is considered the best architectural strategy in developing the AIO system. It offers a conceptual role where the user interfaces with one central server, which allows for security and storage for bookmarks and data processing. While the client provides a user interface, the request and maintenance of data are governed by the server. For the purposes of this system, such a design will provide scalability, security, and ease of maintenance.
- Interface Design Rules: The AIO system adheres to requisite interface design rules, including accessibility, ease of use, and clarity. Users receive guidance through interaction with lucid error notifications along with system feedback, while structured layouts and recognizable icons with standardized components uphold consistency. Accessibility standards attend to assorted user needs. Users can customize themes, categories, and other interface elements. These prescribed guidelines yield a more efficacious and pleasurable system utilization.
- Consistency: The application maintains a unified layout, color palette, and typography, coupled with custom icons or buttons when it performs similar operations, thus ensuring a predictable yet coherent user experience.

- Usability: The system's utility addresses a thorough array of users' demands. Attributes like shortcuts aid experts, and interpretive constituents serve novices. Considering this composition, the interface remains operational and accessible.
- Accessibility: Adhering to accessibility standards, along with offering multiple communication and navigation methods, ensures system engagement for users with disabilities.
- Feedback: Immediate system responses apprise users concerning their actions because clear notifications as well as helpful guidelines are provided through illustrative symbols coupled with textual alerts.
- Error handling: For minimizing errors, the interface is designed with the constraints and controls. Validation messages will help users fix issues efficiently. Visual cues will be provided also when an error occurs.
- User control: Customizing themes along with categories gives users complete control over the interface. Besides UI elements, users personalize them according to their preferences.

As shown in Fig. 7, the proposed system interface presents the main features of AIO platform.

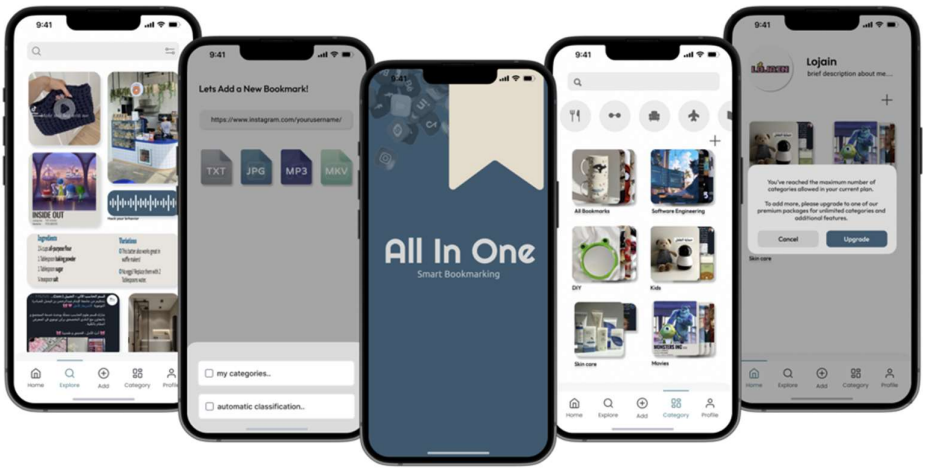

Figure 6. Interface Design.

D. Software Test Plan

This section is dedicated to the AIO platform to establish a detailed plan for system testing. It identifies the items to be tested to ensure that the system's modules, data, databases, and interfaces comply with the GUI requirements. It also specifies the functional requirements that will be tested as well as those that will not. The section includes a detailed explanation of the testing components and defines the suspension, resumption, and approval criteria before presenting the pass/fail standards. Furthermore, it describes the testing process, including tasks, deliverables, responsibilities, resources, and schedule, as well as environmental requirements such as hardware, software, tools, risks, and assumptions [18]. The purpose of STP is to clarify the methods for executing all types of tests mentioned in the scope. Depending on the element and type, multiple tests will be conducted such as component, integration, conversion, job stream, interface, security, recovery, performance, regression, acceptance, and beta testing [18].

Test Items

- Program Modules: Program modules are the division of the system into smaller, independently tested units that have been integrated recently with the AIO platform [19].
- Job Control Procedures: Functionality, interfaces, performance, and security are sequentially tested after each component is tested individually and followed by system testing. Moreover, acceptance and beta testing are carried out to find defects that might still exist [18].
- User Procedures: All user actions within AIO will be tested to ensure smooth and efficient operation.
- Operator Procedures: It is a structured testing process that verifies functionality, usability, compatibility across systems and browsers, performance, and security to ensure readiness in the production environment.

Features to Be Tested

- Common Features: Login, Retrieve Password.
- Manager Features: Manage Customers, View Dashboard, Delete Inappropriate Bookmarks.

- Customer Features: Registration (Create Account), Add Bookmark, Edit Bookmark, Delete Bookmark, Categorize Bookmarks, Publish Bookmarks, View Personal Bookmarks, View Other Customers' Bookmarks, Create Category, Delete Category, Edit Category.

Features Not to Be Tested
Customer Contact ,Edit Profile, Review Page.

Test Types
Software testing is the work of investigating the software elements to find the variances between the desired and the actual conditions and to assess the characteristics of the AIO platform. The AIO testing procedure comprises such tests, as:

- Component Testing: Every part of the AIO system is tested separately to verify that it works as expected. When bugs are discovered early it is much simpler and quicker to fix the problems and it also saves money. Completion criteria require each component to produce the expected outcome before moving on to integration testing [19].
- Integration Testing: Conducted after all components have been individually verified. The goal is to ensure that the modules interact and function together smoothly. Testing is performed incrementally to verify the validity of each integration. Completion criteria confirm that the entire system works as expected, covering both the manager's homepage and the customer homepage [18].
- Conversion Testing: Ensures that the transformation of data from one format to another preserves integrity and accuracy. Any change in the database must be reflected correctly in the user interfaces. This may be achieved either manually or by automated scripts. Completeness requires that data conversion be accomplished without any violations or loss [20].
- Job Stream Testing: The test involves running a series of jobs in a predefined order, where the output of one is the input of another. The aim of this test is to validate that dependencies between processes do not cause failures and that workflows operate seamlessly [21].
- Interface Testing: Detects any flaws in system interactions and verifies that all user and manager operations (such as login, registration, password retrieval, dashboard access, and bookmark handling) function correctly. Errors are displayed properly to guide the user when inputs are invalid [18].
- Security Testing: It ensures that the system keeps the information that the user considers confidential, and at the same time, it does not allow access to anyone who is not authorized. Examples of specific tests can be the security of payments and transactions, the validity of password resets via secure email links, and the requirement of authentication before system resources get access. The main aim is to have security that is at a high level and can meet the user's expectations [18].
- Recovery Testing: Recovery testing is a test of the system that is carried out to check if the system can come back from a failure situation, which may be a crash or data corruption. Recovery is the process of restoring the system to its last stable state and re-executing all operations until the point of failure. According to the definition of done, the data must be accurate and easily accessible after the recovery operation [18] [22].
- Performance Testing: This test aims for investigates the performance of the application under various loads. The properties such as response time, throughput, availability, portability, and scalability are measured in this type of testing. For example, the system must handle a maximum number of transactions per second, support concurrent users without crashing, maintain speed across different operating systems, and perform regular database backups [18].
- Regression Testing: This is the process of testing that comes after the change or upgrade of a system with the aim of showing that the change has not led to the creation of any new bugs. Both minor changes (e.g., editing a few lines of code) and major changes (e.g., adding or replacing modules) are verified. With the help of automated regression tests, a large amount of time can be saved, and the continuity of business can be maintained even if there are rapid changes in software [18].
- Acceptance Testing: Ensures that the system is in line with customer needs and expectations. It is carried out as User Acceptance Testing (UAT), where end-users validate the system's functional and non-functional requirements. A system without errors that passes acceptance criteria will be released [23].
- Beta Testing: This test is performed by a selected group of users in a real-world environment before the final release. The test aims to identify the issues that were not detected during the earlier phases and to gather feedback on usability and performance. A stable version that satisfies all requirements is approved for release [24].

VI. Conclusion and Future Works

AIO platform provides an efficient and effective solution for managing and organizing scattered bookmarks from different social media applications together into a single, unified platform. Its ultimate aim is to maximize the utilization of digital content the individuals are exposed to by offering saving, categorization, search, and sharing features that enhance productivity and improve the user's experience. In the future, the system can be enhanced by adding AI-based recommendations, highlighting online shopping options that are linked with the bookmarks, and adding smart assistants such as intelligent bots and technical support services to help improve the overall user experience.

Author Contribution

All authors contributed equally to the conceptualization, system design, data collection, analysis, and manuscript preparation, and all authors reviewed and approved the final version of the manuscript.

Acknowledgment

The authors would like to express their sincere gratitude to Dr. Enas El-Sharawy for continuous guidance along with indispensable supervision throughout this research. We also extend our gratitude to Nada B. Alrshidi for her active contribution to all software engineering project stages, as well as for playing a meaningful role early on. We further acknowledge that the Department of Computer Science at Imam Abdulrahman Bin Faisal University supported us, and all participants within the surveys and usability testing offered valuable feedback, which greatly improved the quality of this work.

References

[1] E. Eren, "The impact of digital technologies on public life," AI & Society, vol. 39, no. 2, pp. 543–558, 2024.

[2] P. L. Blanc, "How emerging technologies shape the future of work," Eur. J. Work Organ. Psychol., vol. 33, no. 5, pp. 665–682, 2024.

[3] United Nations Conference on Trade and Development (UNCTAD), Technology and Innovation Report 2025: Inclusive Artificial Intelligence for Development, Geneva, Switzerland, 2025. [Online]. Available: https://unctad.org/publication/technology-and-innovation-report-2025. Accessed: Sep. 19, 2025.

[4] Y. Zhang, M. Liu, W. Chen, and J. Sun, "Social media overload and its impact on decision-making: A behavioral perspective," Behav. Sci., vol. 14, no. 8, p. 636, 2024. [Online]. Available: https://www.mdpi.com/2076-328X/14/8/636

[5] M. J. Eppler and J. Mengis, "Information overload: An overview," City University of London Open Access Repository, London, U.K., 2020. [Online]. Available: https://openaccess.city.ac.uk/id/eprint/23544/

[6] DataReportal, "Global social media statistics: Social media users," 2025. [Online]. Available: https://datareportal.com/social-media-users. Accessed: Sep. 19, 2025.

[7] J. Hu and S. M. Noor, "Why we share: A systematic review of knowledge-sharing intentions on social media," Behav. Sci., vol. 14, no. 8, p. 636, 2024. [Online]. Available: https://www.mdpi.com/2076-328X/14/8/636

[8] O. Bergman, S. Whittaker, and J. Schooler, "Out of sight and out of mind: Bookmarks are created but not used," J. Librariansh. Inf. Sci., vol. 53, no. 2, pp. 338–348, 2021. [Online]. Available: https://doi.org/10.1177/0961000620949652

[9] Y. Wang, X. Zhao, T. Xu, and X. Wu, "Autofield: Automating feature selection in deep recommender systems," in Proc. ACM Web Conf. (WWW '22), Lyon, France, Apr. 2022, pp. 1977–1986.

[10] M. M. Y. Khalil, Q. Wang, B. Chen, and W. Wang, "Cross-modality representation learning from transformer for hashtag prediction," J. Big Data, vol. 10, no. 1, p. 148, Sep. 2023. [Online]. Available: https://doi.org/10.1186/s40537-023-00824-2

[11] B. Choi, J. Arguello, R. Capra, and A. R. Ward, "Orgbox: A knowledge representation tool to support complex search tasks," in Proc. Conf. Human Information Interaction and Retrieval, 2021, pp. 219–228.

[12] L. Eric et al., "Bewilder: Handling web resource complexity in online learning and research," Dept. Comput. Sci., Univ. of British Columbia, Vancouver, BC, Canada, Tech. Rep., 2020. [Online]. Available: https://www.cs.ubc.ca/~tmm/courses/547-20/projects/eric/final-report.pdf

[13] D. Hu and S. W. Lee, "Scrapbook: Screenshot-based bookmarks for effective digital resource curation across applications," in Proc. 35th Annu. ACM Symp. User Interface Software and Technology (UIST '22), Oct. 2022, pp. 1–13. [Online]. Available: https://doi.org/10.1145/3526113.3545678

[14] U. M. Khan, Link App and Shortener, iOS application, ver. 1.4.2, Apple App Store, Aug. 10, 2025. [Online]. Available: https://apps.apple.com/sa/app/link-app-and-shortener/id6467019637

[15] Pinterest Inc., "Pinterest," 2010. [Online]. Available: https://www.pinterest.com/. Accessed: Sep. 14, 2025.

[16] Hammel, "Hammel File Manager," 2023. [Online]. Available: https://www.hammel.in/. Accessed: Sep. 19, 2025.

[17] Raindrop.io, "Raindrop.io—All-in-one bookmark manager," 2025. [Online]. Available: https://raindrop.io. Accessed: Sep. 13, 2025.

[18] I. Sommerville, Software Engineering, Global ed. Sydney, Australia: Pearson, 2015.
[19] Blueprint—Blog by Tiny, "Modular programming: Definitions, benefits, and predictions," 2024. [Online]. Available: https://www.tiny.cloud/blog/modular-programming-principle/
[20] M. Anwar, "What is data conversion: Techniques, tools, and best practices," 2022. [Online]. Available: https://www.astera.com/type/blog/data-conversion/. Accessed: Dec. 7, 2024.
[21] IBM, "IBM workload automation 10.2.1," 2024. [Online]. Available: https://www.ibm.com/docs/en/workload-automation/10.2.1?topic=objects-job-stream-definition. Accessed: Dec. 10, 2024.
[22] IEEE Standard Glossary of Software Engineering Terminology, IEEE Std. 610.12-1990, 1990.
[23] GeeksforGeeks, "Acceptance testing—Software testing," 2019. [Online]. Available: https://www.geeksforgeeks.org/acceptance-testing-software-testing/. Accessed: Dec. 8, 2024.
[24] A. Carpenter, "What is user acceptance testing (UAT)?" 2022. [Online]. Available: https://www.codecademy.com/resources/blog/what-is-user-acceptance-testing/. Accessed: Dec. 8, 2024.