Design and Build a Web Service for Financial Technology Dashboard at PT. Mitra Kasih Perkasa with React JS

Yehuda Joy Muljanto[1], Ridwan Sanjaya[2], Albertus Dwiyoga Widiantoro[3]
[1,2,3] Information Systems Department, Faculty of Computer Science
Soegijapranata Catholic University, Indonesia
[1]19n40003@student.unika.ac.id

Abstract— PT Mitra Kasih Perkasa or MKP is a financial technology or fintech company, as well as a system integrator. MKP processes thousands, even millions, of financial transaction data every day, requiring efficient data management through the use of a dashboard. Information from the dashboard is used to monitor business processes and improve company performance. This research focuses on the development of efficient and effective web services for financial technology dashboards, integrating attractive and user-friendly user interfaces (UI/UX). Additionally, this research will discuss how to access web services using React JS, a popular JavaScript framework for responsive user interface development. The research will also outline the steps in building a financial technology dashboard that can help PT. Mitra Kasih Perkasa efficiently and effectively manage financial information using modern and innovative React JS technology.

Keywords— API, dashboard, financial technology, fintech, UI/UX, web service.

I. INTRODUCTION

The development of information technology has had a significant influence on various aspects such as social, cultural, and economic [1]. Financial technology (fintech) comes in various forms, such as crowdfunding, microfinancing, digital payment services, and many more [2]. In practice, fintech manages thousands or even millions of transaction data every day, and current information technology enables the management of large data sets [3]. A dashboard is an application that presents important information in a single full screen to facilitate users in reading and analyzing data [4][5].

Therefore, this research aims to develop a web service for financial technology dashboard at PT. Mitra Kasih Perkasa using React JS with a DevOps approach. PT. Mitra Kasih Perkasa has previously developed a transaction dashboard using the Laravel framework, but this research will migrate it into a web application dashboard format with a DevOps approach to ensure ease of improvement and changes and strengthen the company's position in facing market and industry changes [6][7].

This research will focus on the frontend software development of the apps2pay dashboard using React JS technology, with the goal of providing easily understandable data visualization, effective yet informative design, and features to add, update, and delete data. React JS is a library used to create user interfaces for websites or web applications [8]. JavaScript is a client-side programming language, which means the code is executed in the user's browser, not on the server [9].

User interface/user experience (UI/UX) refers to the appearance of a website or application, designed to be as attractive as possible to achieve user satisfaction [10].

REST (Representational State Transfer) API is a set of architectural principles and guidelines for creating web services [10]. RESTful APIs use standard HTTP methods such as GET, POST, PUT, and DELETE to interact with resources [11][12].

JSON or JavaScript Object Notation is a lightweight data interchange format widely used in web development to send data between server and client applications [13].

Postman is an API client tool that facilitates API development, testing, and management for developers. This tool allows users to create request collections and organize them into folders for efficient management [14].

II. METHOD

Figure 1 illustrates the development method used in this research, which is Rapid Application Development (RAD), with iterative development, prototyping, and feedback.

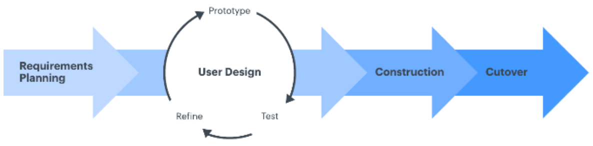


Figure 1. RAD Flowchart

The Technology Acceptance Model (TAM) testing model will be used to understand the factors influencing user acceptance of technology, including Social Influence (SI), Performance Expectancy (PE), Effort Expectancy (EE), Facilitating Conditions (FC), and Behavioral Intention (BI). Statistical tests will be conducted using data from the distributed questionnaire, and the test results will be used as a reference in the analysis of the dashboard development results. The questionnaire will be based on the model and hypotheses for testing the intention of new applications with variables such as usefulness, ease of use, enjoyment, and device support.

Table 1 presents the list of questionnaire questions used to discuss the results of the web application dashboard creation for apps2pay..

Table 1. List of questionnaire questions

	Questions	Answers
1	My colleagues and supervisors often discuss the benefits of using the Apps2pay web dashboard application in achieving work goals and performance targets.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
2	I receive influence from my colleagues and superiors to use the Apps2pay web dashboard application in achieving work goals and performance targets.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
3	The use of the Apps2pay web dashboard application makes it easier for me to achieve work goals and performance targets.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
4	The use of the Apps2pay web dashboard application improves my efficiency and performance in achieving work goals and targets.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
5	The use of the Apps2pay web dashboard application provides support for me in making decisions to achieve work goals and targets.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
6	I believe that it does not take long to learn and use the Apps2pay web dashboard application.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
7	I believe that it does not require extensive knowledge to learn and use the Apps2pay web dashboard application.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree

	Questions	Answers
8	I believe that it does not require extensive knowledge to learn and use the Apps2pay web dashboard application.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
9	I am confident that the company has provided sufficient infrastructure to support the use of the Apps2pay web dashboard application.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
10	I am committed to using the Apps2pay web dashboard application regularly in order to achieve work goals and performance targets.	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree
11	Saya berencana untuk terus menggunakan aplikasi web dashboard Apps2pay di masa depan	Description: 1 = Strongly Disagree 2 = Disagree 3 = Neutral 4 = Agree 5 = Strongly Agree

III. RESULTS AND DISCUSSION

A. RESULT

Figure 2 displays the process flow in the Apps2pay web dashboard application, starting from the user login step, accessing the dashboard, performing CRUD operations on data, accessing other parts of the application, and finally logging out as the final step in using the application. Each of these processes utilizes API services to execute web functions that are more intuitive and interactive.

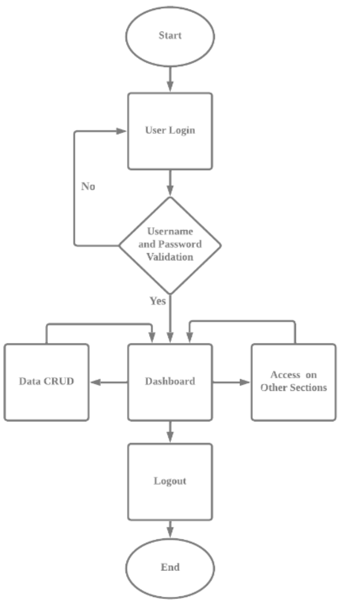
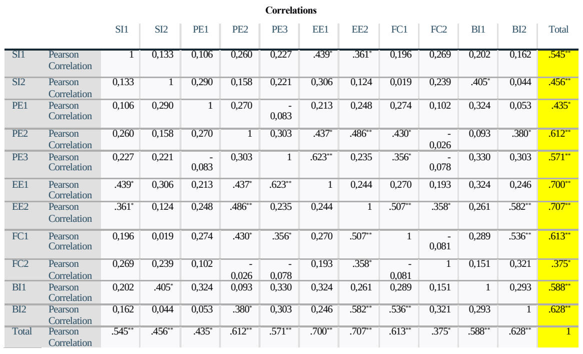


Figure 2. Flowchart of web dashboard application

In designing the Apps2pay web dashboard application, the approach taken is to utilize the existing dashboard as a reference for designing the new user interface (UI/UX) with the aim of saving time and effort and ensuring consistency between the new dashboard and the existing one, making the design process easier.

From the results of Pearson correlation testing on the data obtained from 31 respondents, as shown in Table 2, it can be concluded that there is a significant positive correlation between variable X and variable Y, with a p-value of 0.005 which is smaller than the predetermined alpha level (α) [15].

Table 2. Results of Pearson correlation test



Correlations

		SI1	SI2	PE1	PE2	PE3	EE1	EE2	FC1	FC2	BI1	BI2	Total
SI1	Pearson Correlation	1	0,133	0,106	0,260	0,227	.439*	.361*	0,196	0,269	0,202	0,162	.545**
SI2	Pearson Correlation	0,133	1	0,290	0,158	0,221	0,306	0,124	0,019	0,239	.405*	0,044	.456**
PE1	Pearson Correlation	0,106	0,290	1	0,270	-0,083	0,213	0,248	0,274	0,102	0,324	0,053	.435*
PE2	Pearson Correlation	0,260	0,158	0,270	1	0,303	.437*	.486**	.430*	-0,026	0,093	.380*	.612**
PE3	Pearson Correlation	0,227	0,221	-0,083	0,303	1	.623**	0,235	.356*	-0,078	0,330	0,303	.571**
EE1	Pearson Correlation	.439*	0,306	0,213	.437*	.623**	1	0,244	0,270	0,193	0,324	0,246	.700**
EE2	Pearson Correlation	.361*	0,124	0,248	.486**	0,235	0,244	1	.507**	.358*	0,261	.582**	.707**
FC1	Pearson Correlation	0,196	0,019	0,274	.430*	.356*	0,270	.507**	1	-0,081	0,289	.536**	.613**
FC2	Pearson Correlation	0,269	0,239	0,102	-0,026	-0,078	0,193	.358*	-0,081	1	0,151	0,321	.375*
BI1	Pearson Correlation	0,202	.405*	0,324	0,093	0,330	0,324	0,261	0,289	0,151	1	0,293	.588**
BI2	Pearson Correlation	0,162	0,044	0,053	.380*	0,303	0,246	.582**	.536**	0,321	0,293	1	.628**
Total	Pearson Correlation	.545**	.456**	.435*	.612**	.571**	.700**	.707**	.613**	.375*	.588**	.628**	1

Based on the results of Pearson correlation test, it was found that all Pearson correlation values obtained exceed the value of 0.3440 according to the r calculated table.

Reliability testing of data using Cronbach's alpha method in Table 3 shows a good level of reliability, with an alpha value of 0.792 and N of items totaling 11. An alpha value exceeding 0.6 indicates high consistency among the items in the instrument, and testing conducted on a large number of items indicates that the data is reliable and valid as a basis for data analysis.

Table 3. Table of Reliability Test Results

Reliability Statistics

Cronbach's Alpha	N of Items
.792	11

B. DISCUSSION

In the login process flowchart in Figure 3, users are prompted to enter their account information such as username and password. If the entered information is correct, the system will verify the data and the login will be successful. However, if the information is incorrect, the user will be returned to the login step to try again.

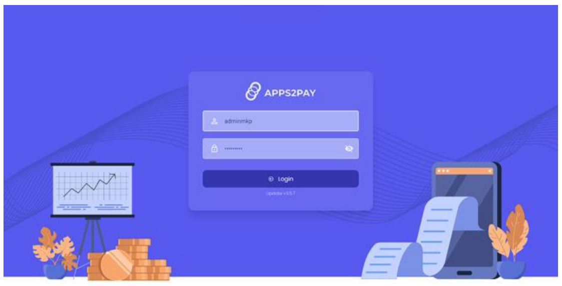


Figure 3. Login Page

After successful login, the user will be directed to the dashboard page, which serves as the main page in the web application dashboard. The dashboard page in Figure 4 contains summarized data, charts, or other relevant information accessed through APIs that are executed using various functions when the dashboard page is accessed.

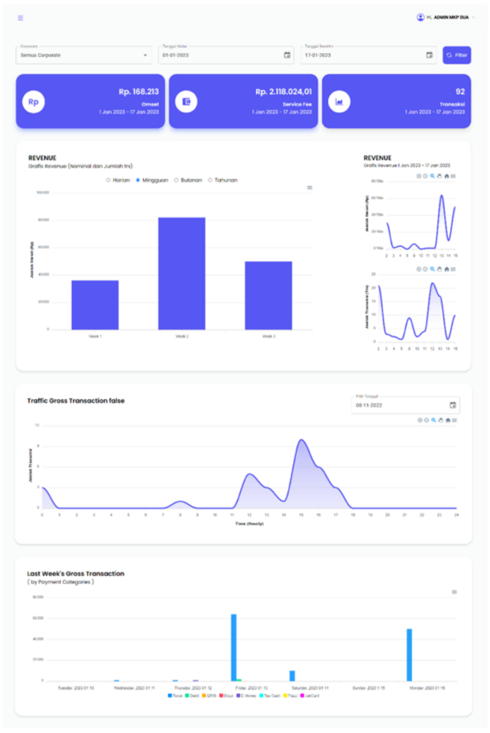
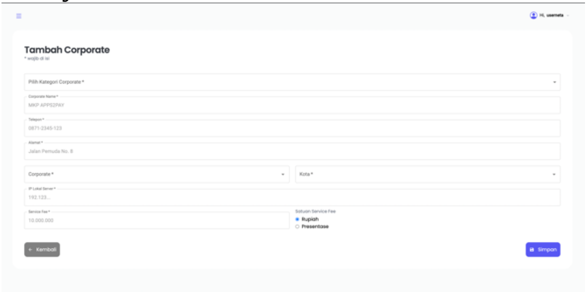


Figure 4. Dashboard Page

After successful login and accessing the dashboard, the CRUD (Create, Read, Update, Delete) data process occurs in the web application dashboard. This process requires user role information that is stored during the login process as an authentication factor, and this variable is processed into boolean data for user access validation in performing CRUD operations on data.

In this section, users can add new data to the system through a form that is filled with the required information, as shown in Figure 5. After the user fills out the form, the data is sent to the system for processing according to the defined business logic or process, such as storing the data in the database, updating the page view, or performing business processes as needed by the system.



Hi, usernata

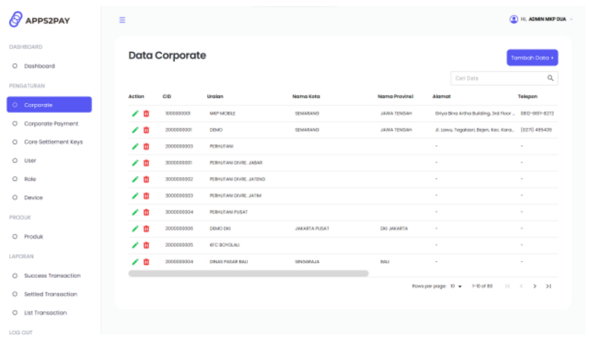
Tambah Corporate
* wajib di isi

Pilih Kategori Corporate *	
Corporate Name *	MKP APPS2PAY
Telepon *	0871-2345-123
Alamat *	Jalan Pemuda No. 8
Corporate *	
Kota *	
IP Lokal Server *	192.123...
Service Fee *	10.000.000
Satuan Service Fee	[✓ Rupiah] Presentase

Kembali   Simpan

Figure 5. Add Data Form

At this stage, users can view existing data in the system through a display as shown in Figure 6, in the form of tables, charts, or other relevant visualizations in the context of the web dashboard application. This display of existing data allows users to access relevant information, perform data analysis, and make decisions based on the displayed data for necessary actions within the web dashboard application.



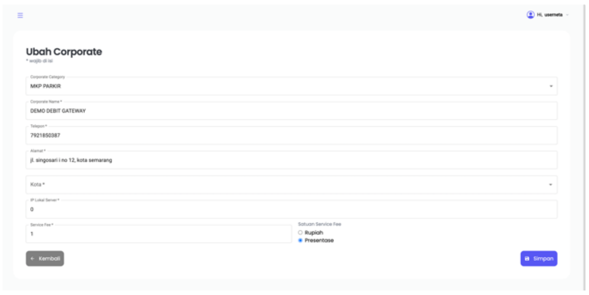
Data Corporate

Action	CID	Uraian	Nama Kota	Nama Provinsi	Alamat	Telepon
	1000000001	MKP MOBILE	SEMARANG	JAWA TENGAH	Griya Bina Artha Building, 3rd Floor ...	0812-6511-8272
	2000000001	DEMO	SEMARANG	JAWA TENGAH	Jl. Lawu, Tegalsari, Bejen, Kec. Kara...	(0271) 495439
	2000000003	PERHUTANI			-	-
	3000000001	PERHUTANI DIVRE. JABAR			-	-
	3000000002	PERHUTANI DIVRE. JATENG			-	-
	3000000003	PERHUTANI DIVRE. JATIM			-	-
	3000000004	PERHUTANI PUSAT			-	-
	2000000006	DEMO DKI	JAKARTA PUSAT	DKI JAKARTA	-	-
	2000000005	KFC BOYOLALI			-	-
	2000000004	DINAS PASAR BALI	SINGARAJA	BALI	-	-

Rows per page: 10 1-10 of 80

Figure 6. Page displaying data table

This process allows users to update existing data in the system, such as editing or correcting information that is already stored in the database. User interaction with the application interface is in the form of a form, as shown in Figure 7, which is used to modify the values of existing data within the web dashboard application.

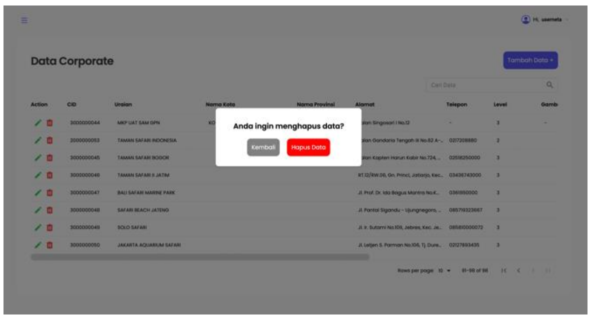


Ubah Corporate
* wajib di isi

Corporate Category	MKP PARKIR
Corporate Name *	DEMO DEBIT GATEWAY
Telepon *	7921850387
Alamat *	jl. singosari i no 12, kota semarang
Kota *	
IP Lokal Server *	0
Service Fee *	1
Satuan Service Fee	Rupiah [✓ Presentase]

Figure 7. Data edit form

In this process, users can delete unnecessary data from the system through the application interface. The deleted data will be removed from the database after the user confirms the deletion. Validation is performed through the application interface that has been designed with appropriate user interface (UI) and user experience (UX), as seen in Figure 8.



Data Corporate

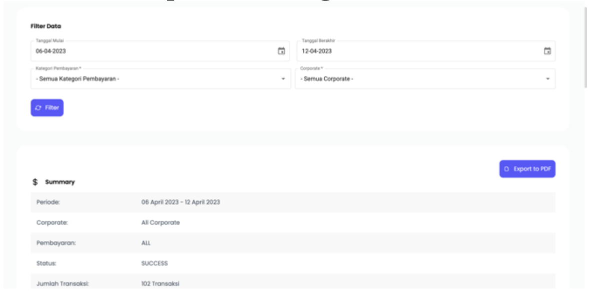
Anda ingin menghapus data?

Action	CID	Uraian	Nama Kota	Nama Provinsi	Alamat	Telepon	Level	Gamb
	3000000044	MKP UAT SAM GPN	KO		Jalan Singosari I No.12	-	3	-
	3000000053	TAMAN SAFARI INDONESIA			Jalan Gandaria Tengah III No.62 A-...	0217208880	3	
	3000000045	TAMAN SAFARI BOGOR			Jalan Kapten Harun Kabir No.724, ...	02518250000	3	
	3000000046	TAMAN SAFARI II JATIM			RT.12/RW.06, Gn. Princi, Jatiarjo, Kec...	03436743000	3	
	3000000047	BALI SAFARI MARINE PARK			Jl. Prof. Dr. Ida Bagus Mantra No.K...	0361950000	3	
	3000000048	SAFARI BEACH JATENG			Jl. Pantai Sigandu - Ujungnegoro, ...	085719323667	3	
	3000000049	SOLO SAFARI			Jl. Ir. Sutami No.109, Jebres, Kec. Je...	085810000072	3	
	3000000050	JAKARTA AQUARIUM SAFARI			Jl. Letjen S. Parman No.106, Tj. Duren...	02127893435	3	

Rows per page: 10 91-98 of 98

Figure 8. Data deletion validation

In addition to CRUD (Create, Read, Update, Delete) data features, there are other features in the dashboard application that allow users to export data into a PDF file, as exemplified in Figure 9.



Filter Data

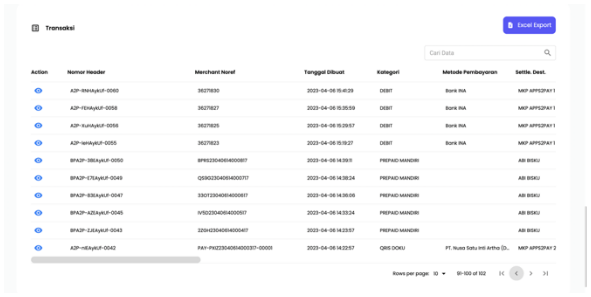
Tanggal Mulai	06-04-2023
Tanggal Berakhir	12-04-2023
Kategori Pembayaran *	- Semua Kategori Pembayaran -
Corporate *	- Semua Corporate -

$ Summary

Periode:	06 April 2023 - 12 April 2023
Corporate:	All Corporate
Pembayaran:	ALL
Status:	SUCCESS
Jumlah Transaksi:	102 Transaksi

Figure 9. Export PDF Menu Page

In Figure 10, which is depicted in the visual representation or screenshot, a web page or interface is displayed. This page provides functionality that allows users to export data from the system in the form of an Excel file. This export feature allows users to save data in a structured and organized format that can be easily opened and manipulated using Microsoft Excel or other spreadsheet software.



Transaksi

Action	Nomor Header	Merchant Noref	Tanggal Dibuat	Kategori	Metode Pembayaran	Settle. Dest.
	A2P-RNHAykUf-0060	36271830	2023-04-06 15:41:29	DEBIT	Bank INA	MKP APPS2PAY 1
	A2P-FEHAykUf-0058	36271827	2023-04-06 15:35:59	DEBIT	Bank INA	MKP APPS2PAY 1
	A2P-XuHAykUf-0056	36271825	2023-04-06 15:29:57	DEBIT	Bank INA	MKP APPS2PAY 1
	A2P-leHAykUf-0055	36271823	2023-04-06 15:19:27	DEBIT	Bank INA	MKP APPS2PAY 1
	BPA2P-38EAykUf-0050	BPRS23040614000617	2023-04-06 14:39:11	PREPAID MANDIRI		ABI BISKU
	BPA2P-E7EAykUf-0049	QS9G23040614000717	2023-04-06 14:38:24	PREPAID MANDIRI		ABI BISKU
	BPA2P-B3EAykUf-0047	33OT23040614000617	2023-04-06 14:36:06	PREPAID MANDIRI		ABI BISKU
	BPA2P-AZEAykUf-0045	IV5D23040614000517	2023-04-06 14:33:24	PREPAID MANDIRI		ABI BISKU
	BPA2P-ZJEAykUf-0043	22GH23040614000417	2023-04-06 14:23:57	PREPAID MANDIRI		ABI BISKU
	A2P-nIEAykUf-0042	PAY-PXIZ23040614000317-00001	2023-04-06 14:22:57	QRIS DOKU	PT. Nusa Satu Inti Artha (D...	MKP APPS2PAY 2

Rows per page: 10 91-100 of 102

Figure 10. Export Excel Menu Page.

IV. CONCLUSION

In the process of researching and developing web applications that utilize web services and React JS, it is essential to prioritize user convenience by using existing dashboards as a reference for designing effective and user-friendly UI/UX interfaces. This means considering how users will interact with the application and ensuring that the design is intuitive and easy to use.

Collaboration between UI/UX design and web services is crucial in creating a well-rounded application. By declaring functions and integrating appropriate APIs, the UI/UX design can work in tandem with web services to provide a seamless user experience. This may involve designing UI components that interact with the web service APIs to fetch, update, or delete data, and ensuring that the data displayed in the UI is consistent with the data in the web service.

When accessing web services in a React JS application, the first step is to establish communication with the desired service through the appropriate API. This may involve making HTTP requests, handling responses, and managing error cases. Once the communication is set up, the declared functions can be utilized within React JS components to interact with the web service and perform desired actions, such as displaying data in the UI, submitting form data, or triggering other API calls.

In the context of building a financial technology dashboard using React JS, the design of the UI, functions, and web service APIs all play separate but crucial roles. The UI should be visually appealing and provide a seamless user experience, while the functions should be designed to perform the necessary tasks related to financial data and transactions. The web service APIs should be robust and secure, providing the necessary data and functionality for the dashboard to function properly.

By considering the collaboration between UI/UX design and web services, and leveraging the capabilities of React JS to interact with web services, a functional and visually appealing financial technology dashboard can be developed, providing users with a powerful and user-friendly tool for managing financial data and transactions.

REFERENCES

[1] P. Gupta and T. M. Tham, *Fintech: the new DNA of financial services*. Boston: Walter de Gruyter Inc, 2019.

[2] S. Agarwal and Y. H. Chua, “FinTech and household finance: a review of the empirical literature,” *China Finance Rev. Int.*, vol. 10, no. 4, pp. 361–376, 2020, doi: 10.1108/CFRI-03-2020-0024.

[3] J. Klaas, *Machine Learning for Finance: principles and practice for financial insiders*. in Expert insight. Birmingham Mumbai: Packt Publishing, 2019.

[4] N. H. Riche, C. Hurter, N. Diakopoulos, and S. Carpendale, Eds., *Data-driven storytelling*. in A K Peters Visualization Series. Boca Raton, Florida: CRC Press/Taylor & Francis Group, 2018.

[5] C. N. Knaflic, *Storytelling with data: let’s practice!* Hoboken, New Jersey: John Wiley & Sons, Inc, 2019.

[6] Y. Raheja, “Effective DevOps with AWS”.

[7] O. H. Plant, J. van Hillegersberg, and A. Aldea, “Rethinking IT governance: Designing a framework for mitigating risk and fostering internal control in a DevOps environment,” *Int. J. Account. Inf. Syst.*, vol. 45, no. April, 2022, doi: 10.1016/j.accinf.2022.100560.

[8] S. Aggarwal, “Modern Web-Development using ReactJS,” *Int. J. Recent Res. Asp.*, vol. 5, no. 1, pp. 133–137, 2018.

[9] P. Carey and S. Vodnik, *Javascript for web warriors*, Seventh edition. Boston, MA: Cengage Learning, Inc., 2022.

[10] H. Ji, Y. Yun, S. Lee, K. Kim, and H. Lim, “An adaptable UI/UX considering user’s cognitive and behavior in*form*ation in distributed environment,” *Clust. Comput.*, vol. 21, no. 1, pp. 1045–1058, Mar. 2018, doi: 10.1007/s10586-017-0999-9.

[11] N. Yellavula, *Hands-On RESTful Web services with Go - Second Edition*, 2nd edition. Packt Publishing, 2020.

[12] F. Doglio, *REST API Development with Node.js: Manage and Understand the Full Capabilities of Successful REST Development*. Berkeley, CA: Apress, 2018. doi: 10.1007/978-1-4842-3715-1.

[13 P. A. Carter, *SQL Server Advanced Data Types: JSON, XML, and Beyond*. Berkeley, CA: Apress, 2018. doi: 10.1007/978-1-4842-3901-8.

[14] D. Westerveld, *API Testing and Development with Postman*, 1st edition. Packt Publishing, 2021.

[15] D. S. Moore, G. P. McCabe, and B. A. Craig, *Introduction to the Practice of Statistics*, Tenth edition. New York: WH Freeman, 2021.