proceedings

MDPI

Proceeding Paper

Web-Based Parametric Effort Estimation for Mobile Application Development †

Nur Ida Aniza Rusli [1,*], Nur Atiqah Sia Abdullah [2], Fatin Nabila Abd Razak [3] and Nemo Menton Mufriz [2]

[1] Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Cawangan Negeri Sembilan, Kampus Kuala Pilah, Kuala Pilah 72000, Negeri Sembilan, Malaysia
[2] Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Shah Alam 40450, Selangor, Malaysia
[3] Academy of Language Studies, Universiti Teknologi MARA, Cawangan Negeri Sembilan, Kampus Kuala Pilah, Kuala Pilah 72000, Negeri Sembilan, Malaysia
* Correspondence: idaaniza@uitm.edu.my
† Presented at the International Academic Symposium of Social Science 2022, Kota Bharu, Malaysia, 3 July 2022.

Abstract: Estimation methods are continuously being adapted to obtain better and clearer estimations needed to achieve development goals. Some estimation methods were invented before the modern mobile application technology that is currently available. Thus, these methods are unable to cater to the requirements for estimating modern mobile application features. The objective of this paper is to propose a web-based system as a method to estimate the effort and cost of developing a mobile application. The key idea behind this study is to identify cost drivers that can be applied in mobile application development through literature review. From the analysis, 19 cost drivers are found to fit the vision of this study. In addition, this study also seeks to investigate the price range of cost drivers acquired from existing similar systems. The total price range is accumulated, and the mean value of each cost driver is obtained, which is then inserted further into the new estimation metric. The proposed system is then evaluated by comparing the obtained results with six similar systems according to basic user needs requirements in an application. The results demonstrate that the proposed system is a more enhanced cost estimation software that contains more cost driver options, which users can utilize to estimate mobile application development costs.

Keywords: software effort estimation; parametric estimation; mobile application; estimation system

Citation: Rusli, N.I.A.; Abdullah, N.A.S.; Abd Razak, F.N.; Mufriz, N.M. Web-Based Parametric Effort Estimation for Mobile Application Development. *Proceedings* **2022**, *82*, 69. https://doi.org/10.3390/proceedings2022082069

Academic Editor: Mohamad Rahimi Mohamad Rosman

Published: 20 September 2022

1. Introduction

Effort estimation is the procedure carried out to anticipate the most sensible measure of effort required to create or maintain software. Effort estimation is a key project management activity needed for project planning, staff resources estimation, cost estimation, quality control, and benchmarking [1]. Enhancing the estimation techniques available to project managers would encourage more successful control of time and spending plans in software development [2].

The ever-growing need for better functionality and hope for a better way of life has brought forth a whole new mobile application development industry. Despite the accessibility of many versatile applications, software developers create many new applications to fulfill the interest of mobile device users worldwide [1]. This results in developers seeking the most efficient techniques for effort estimates for project plans, cycle designs, spending plans, investment analysis, pricing processes, and bidding rounds. Inappropriate software development effort estimation can result in project failures due to budget overruns and slips in scheduling [1].

Furthermore, existing methods such as Function Point, Object Point, and COSMIC Full Function Point (COSMIC FFP) have limitations, as they are prone to these inclinations: individual experience, political points, resources, time weight, and memory recall [3–5]. In

addition, these estimation methods were invented before the modern mobile application technology available now, and they are most likely unable to cater to current features.

Therefore, new estimation methods are needed to estimate mobile application development efforts to overcome the estimation problems. Therefore, this study proposes a web-based parametric effort estimation system as an option for software developers or other users to estimate the cost of mobile application development. This paper is organized as follows: Section 2 describes the related works, Section 3 illustrates the proposed system, Section 4 summarizes the results, while Section 5 draws the conclusion.

2. Related Works

The following section reviews existing effort estimation models, consisting of mobile application estimation models and effort estimation systems.

2.1. Mobile Application Estimation Models and Systems

The effort estimation of mobile applications is a complex issue, and no specific model or process exists. It has been demonstrated from natural considerations that mobile application development suffers from effort estimation syndrome. Therefore, there have been several attempts in the last few years to address characteristics and techniques in the field of effort estimation across mobile applications.

Shahwaiz et al. [1] proposed a parametric model for assessing the effort necessary to create mobile applications. The regression-based model is measured using information from 161 mobile application characteristics and validated using the k-fold cross-validation method. In addition, the expected precision of this mobile application's particular model is contrasted with the standard precision of the general-purpose COCOMO II model. The correlation result demonstrates that this model is more precise than the COCOMO II model. Initially, there were 16 cost drivers, which resulted in seven categories of effort predictors upon calibration.

Altaleb and Gravell [6] depicted the results of a Systematic Literature Review (SLR) with respect to size estimation and effort models in mobile application development. This is followed by an outline of estimation techniques utilized crosswise over mobile applications gathered from 64 papers and presents the suggested 25 cost drivers.

Accuracy and efficiency are critical factors in ensuring a successful effort estimation model. Thus, many organizations have developed effort estimation systems to support the calculation process. This study reviews seven effort estimation systems specifically for mobile application development, including Estimate My App [7], How Much to Make an App [8], VenturePact [9], BuildFire [10], Otreva [11], Cleveroad [12], and Andreas Ley Calculator [13].

2.2. Comparison of Cost Drivers

This study has identified 40 cost drivers from the literature review in [1,6] and used further to identify the most significant cost drivers for our proposed method.

From the comparison in Table 1, this study includes all the cost drivers, with the majority scoring 3/7 or higher. However, the number of screens characteristic is discarded as a potential factor in the proposed method. This is because the number of screens factor is considered a young research discipline.

Software metric researchers, for example, are still trying to find a range for the number of screens required to classify a mobile app's complexity (either in small, medium, or large) [14,15]. Thus, the general effort costing cannot be derived, as no proper evaluation has been conducted for this factor.

Table 1. Comparison of cost drivers towards seven estimation systems.

Characteristics	Estimate My App	How Much to Make an App	Venture Pact	BuildFire	Otreva	Cleveroad	Andreas Ley Cost Calculator	Total
Function point size	/	-	-	-	-	-	-	1/7
UML diagram	-	-	-	-	-	-	-	0/7
Supported platform type	/	/	/	/	/	/	/	7/7
Supported device	-	-	-	-	-	-	/	1/7
Supported device	-	-	-	-	-	-	/	1/7
Back-end system availability and server config. flexibility	-	-	-	/	-	-	-	1/7
Development team skills	-	-	-	-	-	-	-	0//7
App development flexibility and complexity	-	-	-	-	-	-	-	0//7
Team communication, process, complexity and experience	-	-	-	-	-	-	-	0/7
Push notification	/	-	-	/	/	/	/	5/7
Landscape and portrait mode	-	-	-	-	-	-	-	0/7
Data storage and memory opt. complexity	-	-	/	/	-	-	-	2/7
Number of screens	-	-	/	-	-	/	/	3/7
Number of API parties	/	/	-	/	/	-	-	4/7
Support code reusability	-	-	-	-	-	/	/	2/7
Technology maturity	-	-	-	-	-	-	-	0/7
Battery and optimisation	-	-	-	-	-	-	-	0/7
Connection	-	/	-	/	-	-	/	3/7
Booking and reservation	/	-	-	-	/	-	-	2/7
Calendar and time	/	-	-	/	/	/	-	4/7
Map and localisation	/	-	/	/	/	/	/	6/7
Social sharing	/	/	-	/	/	/	-	5/7
Searching contents	/	-	/	-	/	-	-	3/7
Messaging	/	-	-	/	/	/	-	4/7
Deadline date	/	-	-	-	-	/	-	2/7
Number of functionalities	/	-	-	-	-	-	-	1/7
Registration and login	/	/	/	/	/	/	/	7/7
Chronological list	-	-	-	-	-	-	-	0/7
Number of files, classes, methods, statements, and LOC	-	-	-	-	-	-	-	0/7
Chronological list	-	-	-	-	-	-	-	0/7
File upload	/	-	-	-	-	/	-	2/7
Comment feature	/	/	-	-	-	-	-	2/7
Navigation	-	-	-	-	-	-	/	1/7
Interrupt handling	-	-	-	-	-	-	-	0/7
Security analysis support	/	-	/	/	-	-	/	4/7
Budget for the project	-	-	-	-	-	-	-	0/7
Compatibility with previous version	-	-	-	-	-	-	/	1/7
Multi language support	/	-	-	-	-	-	/	2/7
Media support	/	-	/	/	/	-	/	6/7
Pay process user feedback	/	/	/	/	/	/	/	7/7

Next, a total of two systems applied factors of data storage and memory opt. complexity, support code reusability, booking and reservation, deadline date, file upload, comment feature, and multi-language support. Among these factors, booking and reservation, file upload, comment feature, and multi-language support are included in this study because these factors are considered necessary for mobile business modelling [16–19].

Only one system applied factors such as function point size, supported device, back-end system availability, and server config. flexibility, number of functionalities, navigation, and compatibility with the previous version. From these factors, function point size is considered as an important factor, although only the Estimate My App system applied this factor for mobile app effort estimation. This study also includes this factor because function point size is considered a stable procedure to derive effort estimation [20].

Lastly, none of the selected systems applied cost factors such as UML diagram, development team skills, app development flexibility and complexity, team communication, process, complexity and experience, landscape and portrait mode, technology maturity, battery and power optimization, number of files, classes, methods, statements and LOC, chronological list, hardware access, interrupt handling, and budget for the project. Thus, these criteria are not considered essential and are discarded in the proposed system.

3. Proposed System

This section overviews of the proposed web-based parametric effort estimation for mobile application development.

3.1. Cost Drivers

From the comparison (as shown in Section 2.2), 19 cost drivers are obtained for further use in the proposed system. The estimated costing range for each factor is obtained from the reviewed system. The factors are included in this study to suggest the mobile application category and its respective cost. The estimated cost of each cost driver is shown in Table 2.

Table 2. Cost driver in proposed system.

Cost Driver (Proposed System)	Description	Estimated Cost (USD)
Function Point Size	Small (has around 2–3 key features); Medium (has around 4–7 key features); Large (has around 8–12 key features).	Small (USD 4500); Medium (USD 13,500); Large (USD 22,500)
Supported Platform	iOS (iPhone/iPad app); Android (Android phone/Tablet app)	USD 1200–USD 9600
User Interface Quality and Complexity	MVP (Minimum Viable Product. Very raw but functional); Basic (Still quite basic but pleasing to the eye); Polished (Professional bespoke UI design. May also have some animations and transitions etc.)	MVP (USD 400–USD 5850); Basic (USD 1200–USD 6750); Polished (USD 2400–USD 9600)
Push Notification	Real-time notifications between users, e.g., unread message counts, notifications of editing, etc.	Yes (USD 1350–USD 5000); No (USD 0)
Number of API Parties	Connect to one or more third party services (An information feed that you must incorporate with or an accomplice application); SMS Messaging; (Allow your app to send SMS messages); Phone Number Masking (Calls conducted by your app have masked phone numbers).	Connect to one or more third party services (USD 1350–USD 5000); SMS Message (USD 1800); Phone Number Masking (USD 1800)
Connection	Bluetooth (Use Bluetooth to communicate and transfer data between devices); Wireless (App does not need to connect via internet)	Bluetooth (USD 3840–USD 10,000); Wireless (USD 0);
Booking and Reservation	Managing capacity, choosing the start and end dates etc.	Yes (USD 2250–USD 5250); No (USD 0)
Calendar and Time	Display data in a calendar format	USD 2700–USD 3000;
Map and Localisation	Showing a map with data point e.g., driver locations, venue locations, etc.	Yes (USD 1350–USD 5000); No (USD 0)
Social Sharing	Ability to share pieces of information in a controlled way on social media account to drive engagement	Yes (USD 450–USD 3000); No (USD 0);
Searching and Contents	Users would be able to search content	Yes (USD 1350–USD 4500); No (USD 0);
Messaging	Allowing users within the app to send message to other Account users or group of users	Yes (USD 2250–USD 3750); No (USD 0);
File Upload	Users can upload video, photo content or audio	Yes (USD 1800); No (USD 0);
Registration and Login	Email/Password; Facebook; Twitter; Google	Email/Password (USD 320–USD 9600); Facebook (USD 800–USD 4000) Twitter (USD 800–USD 4000) Google (USD 800–USD 4000)
Comment Feature	Classic forum functionality for account users or simple commenting on information	Yes (USD 2250–USD 3000); No (USD 0);
Security Analysis Support	Security not important (Use Bluetooth to communicate and transfer); Basic Security Measures; Complete Protection (Protection against XSS & SQL Injection)	Security not important (USD 0); Basic Security Measures (USD 400–USD 1300); Complete Protection (USD 2400–USD 7800);
Multi-language Support	Provide support for multiple languages for your app	Yes (USD 1800); No (USD 0)
Media Support	Users are able to modify video, photo content or audio on their gadget (e.g., Filters).	Yes (USD 1800–USD 3770); No (USD 0)
Paying Process User Feedback	You will process ad-hoc or regular payments from users and manage refunds, etc.	Yes (USD 1440–USD 6000); No (USD 0);

3.2. System Interface

Figure 1 shows the proposed user interface for the system. It consists of 19 cost drivers to allow the user to choose based on their requirement specification. Figure 1 shows a part of the main interface for the system.

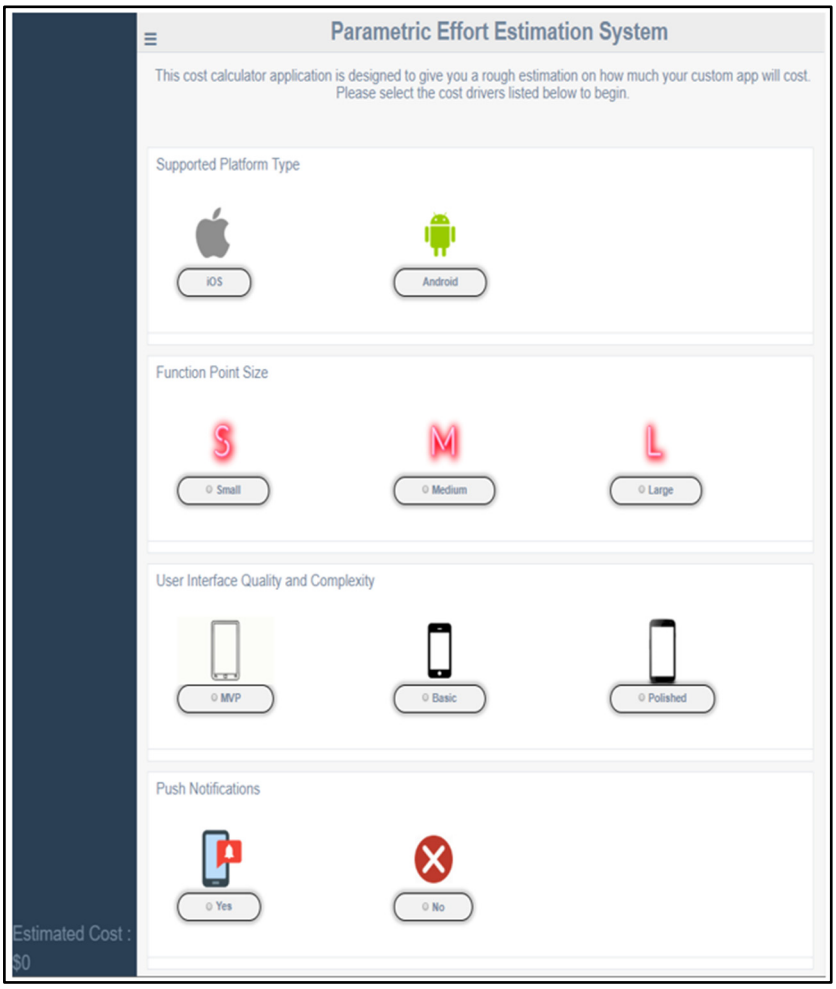

Figure 1. Main interface of proposed system.

There are 19 form groups in the main interface representing 19 cost drivers, as stated in the previous section. Each form group provides two to four options for the user to choose from. These options are types of radio buttons and checkboxes, depending on the cost driver.

Meanwhile, Figure 2 shows an example of a radio button used for the Function Point Size cost driver. This button allows the user to only make one selection for this type of cost driver. The selected option is changed as the user clicks a different option. Every change will deduct the previous value of the cost driver and update the new value of the selected cost driver in the bottom left corner. The total estimated cost will be calculated throughout the 19 cost-driver selections. The estimated cost is displayed at the bottom left of the interface.

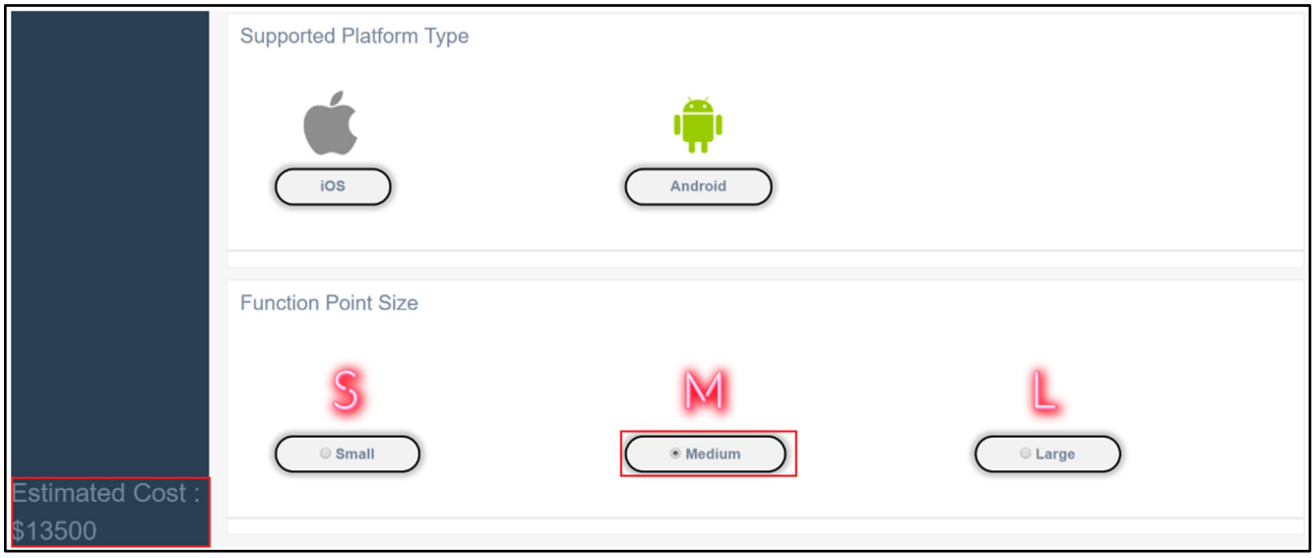

Figure 2. User interface for estimated cost.

Figure 3 shows an example of the checkbox button used for the number of API Parties cost driver. This button allows the user to make more than one selection or remove the selection if the user wishes to do so. For every choice the user makes, the system prompts the total estimated cost on the bottom left corner of the screen. The system will update this value according to the user's selections. The proposed system also provides the tooltips function (refer to Figure 4) to help the user further understand what each cost driver refers to. The tooltip will appear whenever a user hovers over the icons.

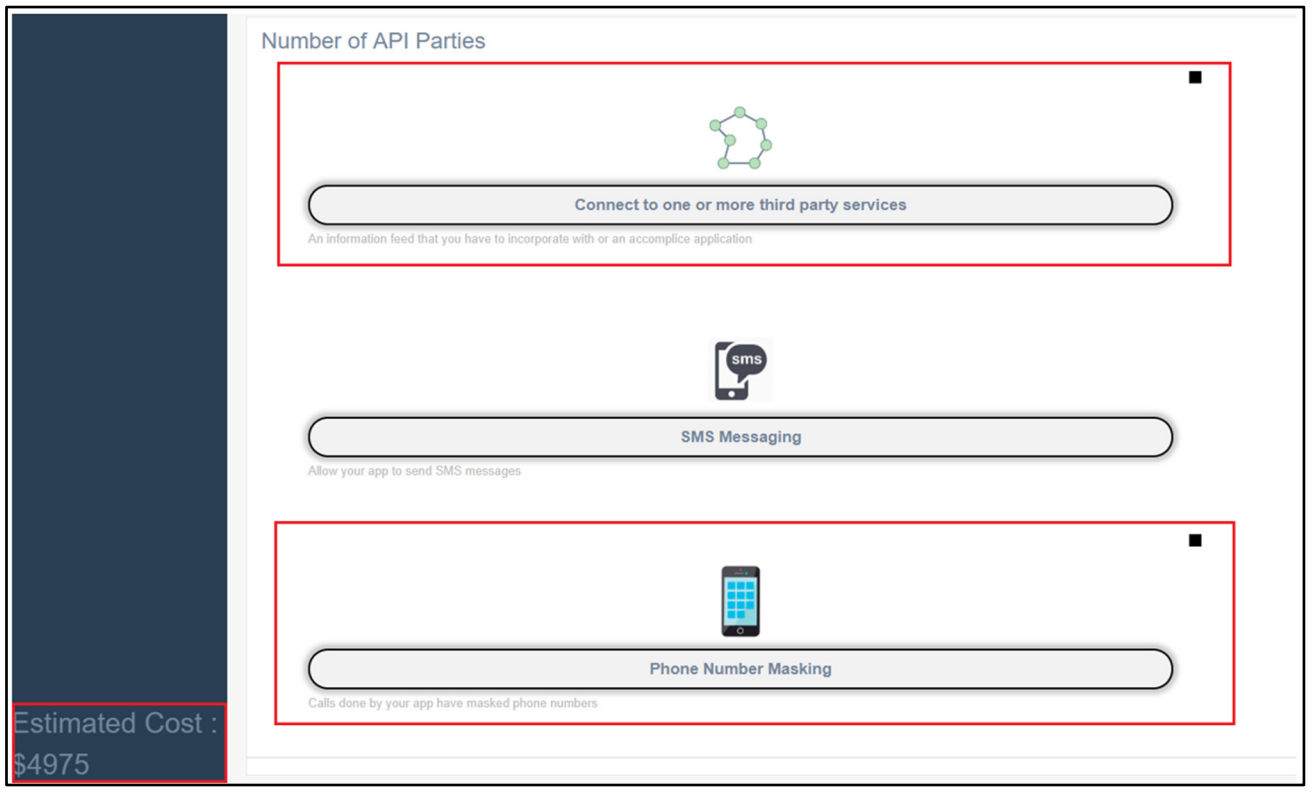

Figure 3. User interface for checkbox function.

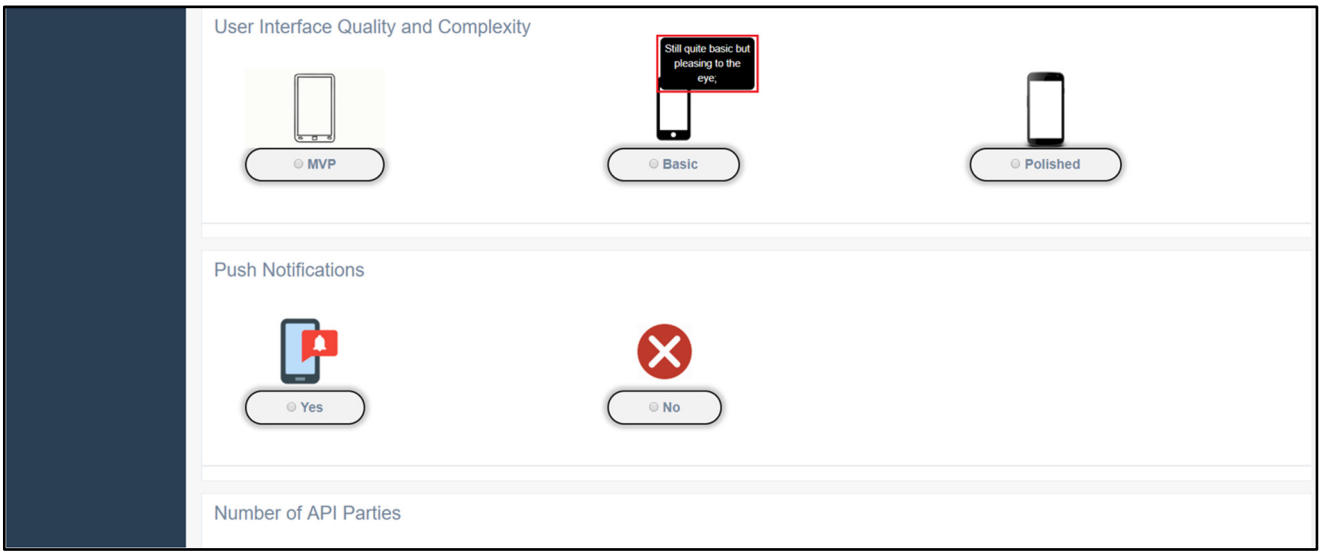

Figure 4. User interface for tooltip function.

4. Result and Discussion

This study evaluated the proposed system's credibility by comparing the proposed system's total value estimate against six other similar systems. The Cleveroad system is excluded from the evaluation since it does not provide any price ranges for the listed cost drivers. As a result, this study omits the Cleveroad system while formulating the cost range of the cost drivers.

Table 3 shows the estimated cost range of similar systems using the standard evaluation criteria. From the result, the percentage of difference between the proposed system and Estimate My App, How Much to Make an App, and Andreas Ley cost calculator are within a range of +10%–+20%.

Table 3. Percentage and difference in terms of cost range.

Title 1	Estimate MyApp	How Much to Make an App	VenturePact	BuildFire	Otreva	Andreas Ley Cost Calculator
	USD 49,500	USD 30,600	USD 24,180	USD 164,868	USD 66,003	USD 31,000
Proposed System	USD 55,300	USD 35,340	USD 18,025	USD 48,550	USD 44,080	USD 36,350
Difference in USD	USD 5800	USD 5340	USD −6155	USD −116,318	USD −21,923	USD 5350
Percentage of Difference	+11.72%	+17.80%	−25.45%	−70.55%	−33.22%	+17.26%

Estimate My App matches almost all the cost drivers (18 out of 19). The system in this study is 11.72% more costly than Estimate My App. The significant difference in percentage collected in the Table 3 is caused by the BuildFire's system having very high charges compared with the other six systems that considered the proposed system's price range. This is justified by seeing that the proposed system is −70.55% lower in cost in comparison with the BuildFire system.

After conducting research into the systematic literature review by Altaleb and Gravel [6], 40 cost drivers were mentioned as important and needed to be accounted for when performing the cost estimation of a mobile application. Moreover, these cost drivers were deemed relevant based on the current needs of e-commerce processes. In this study, only 19 cost drivers were selected for inclusion in this system due to the comparison table that was constructed between similar systems and the systematic literature review.

For future work, the estimated cost range collected in this study may have caused the results to be less appealing. This is because the BuildFire system was included as part in formulating the cost range despite having a much higher rate for their cost driver prices. The BuildFire system, however, is still included in this analysis due to the study's goal

of identifying the most significant cost drivers for current mobile applications. This issue serves as a caution to avoid the future proposed system having extremely high-cost ranges in their cost drivers.

5. Conclusions

This paper has presented a web-based system to estimate mobile application development efforts and costs. The main objective of this research is to identify cost drivers relevant to modern mobile application development. Forty cost drivers were identified from the literature review. However, after analyzing seven systems, only 19 were considered potential factors. In addition, the costing factors were determined using the values offered by the reviewed systems. Furthermore, this study conducted a simple evaluation process to test the functionality of the developed system. This phase was conducted by comparing the total estimated cost of the proposed system with six other similar systems. A comprehensive table that contained the cost range difference in absolute values and percentages was constructed to analyze the results further. In conclusion, the system functions accordingly with a 20–30% significant difference between similar systems.

Author Contributions: Conceptualization, N.I.A.R. and N.A.S.A.; methodology, N.A.S.A. and N.M.M.; software, N.I.A.R., N.A.S.A. and N.M.M.; validation, N.I.A.R., N.A.S.A. and N.M.M.; formal analysis, N.A.S.A. and F.N.A.R.; investigation, N.I.A.R. and N.A.S.A.; resources, N.I.A.R. and N.A.S.A.; data curation, N.I.A.R., F.N.A.R. and N.M.M.; writing—original draft preparation, N.I.A.R. and N.M.M., writing—review and editing, F.N.A.R.; visualization, N.I.A.R. and F.N.A.R.; supervision, N.I.A.R. and N.A.S.A.; project administration, N.I.A.R. and N.A.S.A.; funding acquisition, N.I.A.R. and N.A.S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by MyRA Research Grant Scheme File No.: 600-RMC/GPM LPHD 5/3 (180/2021), under Universiti Teknologi MARA internal grant scheme.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the Ministry of Higher Education of Malaysia (MOHE) for funding the research project under MyRa Grant Scheme. The authors gratefully acknowledge the Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA for supporting the publication of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Shahwaiz, S.; Malik, A.; Sabahat, N. A parametric effort estimation model for mobile apps. In Proceedings of the 19th International Multi-Topic Conference (INMIC), Islamad, Pakistan, 5–6 December 2016; pp. 1–6.
2. Finnie, G.; Wittig, G.; Desharnais, J.M. A comparison of software effort estimation techniques: Using function points with neural networks, case-based reasoning and regression models. *J. Syst. Softw.* **1997**, *39*, 281–289. [CrossRef]
3. Albrecht, A.J. Measuring application development productivity. In Proceedings of the Joint Share/Guide/IBM Application Development Symposium, Monterey, CA, USA, 14–17 October 1979; pp. 83–92.
4. Banker, R.D.; Kauffman, R.J.; Wright, C.; Zweig, D. Automating output size and reuse metrics in a Repository-Based Computer-Aided Software Engineering (CASE) environment. *IEEE Trans. Softw. Eng.* **1994**, *20*, 169–187. [CrossRef]
5. Meli, R.; Abran, A.; Ho, V.T.; Oligny, S. On the applicability of COSMIC-FFP for measuring software throughout its life cycle. In Proceedings of the 11th European Software Control and Metrics Conference, Munich, Germany, 18–20 April 2000.
6. Altaleb, A.R.; Gravell, A.M. Effort estimation across Mobile app platforms using Agile processes: A systematic literature review. *J. Softw.* **2018**, *13*, 242–259. [CrossRef]
7. Oozou. Estimate My App. Available online: https://estimatemyapp.com/ (accessed on 28 October 2020).
8. Crew. How Much to Make an App? Available online: http://howmuchtomakeanapp.com/ (accessed on 28 October 2020).
9. VenturePact. Looking to Build an App? Available online: https://venturepact.com/mobile_app_price_calculator (accessed on 28 October 2020).
10. BuildFire. How Much Will Your Mobile App Project Cost You? Available online: https://buildfire.com/how-much-to-make-a-mobile-app-calculator/# (accessed on 28 October 2020).

11. Otreva. How Much Does It Cost to Build an App? Available online: https://www.otreva.com/calculator/# (accessed on 28 October 2020).
12. Cleveroad. How Much Does It Cost to Make an App? Available online: https://www.cleveroad.com/mobile-app-development-cost-calculator (accessed on 28 October 2020).
13. Ley, A. Cost Calculator for Mobile App Development. Available online: https://www.andreasley.ch/en/costcalculator/ (accessed on 28 October 2020).
14. Iida, N. New Technique for estimating development scale: Implementing the function scale method. *Fujitsu Sci. Tech. J.* **2010**, *46*, 149–157.
15. Rijwani, P.; Jain, S.; Santani, D. Software effort estimation: A comparison-based perspective. *Int. J. Appl. Innov. Eng. Manag.* **2014**, *3*, 18–29.
16. Hebig, R.; Derehag, J.; Chaudron, M.R. Identifying metrics' biases when measuring or approximating size in heterogeneous languages. In Proceedings of the ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, Beijing, China, 22–23 October 2015; pp. 1–4.
17. Gomes, J.V.; Inácio, P.R.; Freire, M.M.; Pereira, M.; Monteiro, P.P. Analysis of peer-to-peer traffic using a behavioural method based on entropy. In Proceedings of the IEEE International Performance, Computing and Communications Conference, Austin, TX, USA, 7–9 December 2008; pp. 201–208.
18. Mendes, E.; Mosley, N.; Counsell, S. Investigating web size metrics for early web cost estimation. *J. Syst. Softw.* **2005**, *77*, 157–172. [CrossRef]
19. Van Den Berg, K.; Dekkers, T.; Oudshoorn, R. Functional size measurement applied to UML-based user requirements. In Proceedings of the 2nd Software Measurement European Forum (SMEF), Rome, Italy, 16–18 March 2005; pp. 69–80.
20. Lavazza, L.; Del Bianco, V. A case study in COSMIC functional size measurement: The rice cooker revisited. In *International Workshop on Software Measurement*; Springer: Berlin/Heidelberg, Germany, 2000; pp. 101–121.