Online Product Monitoring System (OPMS) of DTI

Bryan L. Guibijar

North Eastern Mindanao State University – San Miguel Campus
Brgy. Carromata, 8301 San Miguel, Surigao del Sur, Philippines

Abstract: **The paper examined the model of the data flow diagram in developing Online Product Monitoring (OPMS) for the Department of Trade and Industry (DTI). It is based on the waterfall model in which descriptive research was applied to the existing product monitoring of DTI. Data from existing monitoring of DTI were interpreted and created system data flow diagram (DFD). The data flow diagram is applied in the analysis of software development and examined the top-down method that takes processes the input and output of data flows. Thus, the method developed based on the data flow diagram created will enhance the security and order of the developed online system.**

Keywords: **Data Flow Diagram, Data Aging, Security, Development, Input, and Output data flows**

I. INTRODUCTION

Online development dramatically raises issues, especially in data flow and controlling of data in and out (Li, H., Sun, X., Shi, P., & Lam, H. K., 2015). This issue commonly happens in some open-source online templates in making your site (Hashem, I. A. T., Yaqoob, I., Anuar, N. B., Mokhtar, S., Gani, A., & Khan, S. U., 2015). Lots of students were creating their websites with free hosting and ready-made template (Ravichandran, H., Toohil, R., & Lammi, L., 2014). This leads to catastrophe when comes to data uploading and data downloading. The creation of the Data Flow Diagram in developing an online site will solve the issues. The study aims to provide a standard data flow diagram to enhance Input and Output Data flow (Abedjan, Z., Schulze, P., & Naumann, F., 2014, November).

Data Flow Diagrams are categorized by level, starting with the most basic, levels 0 – 3 (*see figure 1 to figure 3*), DFDs get increasingly complex as the level increases, as you build your data flow diagram, you will need to decide which level your diagram will be. In theory, DFDs could go beyond level 3, but they rarely do, level 3 data flow diagrams are detailed enough that it doesn't usually make sense to break them down further (Bangeter, J., 2017).

II. RESEARCH DESIGN AND METHODS

The study used descriptive research in which a standard Data Flow Diagram was applied. Three-level Data flow Diagram was applied in this research.

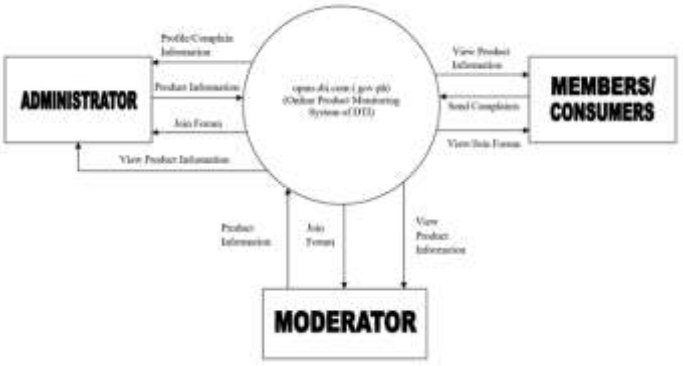


Figure 1. Context level Diagram

The context level diagram shows the level of access rights of different users of the system. Administrator access to all rights like Profile/Complain Information, Product Information, Join Form, and View Product Information. Moderator is a substitute user in case of the absence of the Administrator, the access rights of this user are minimal and that is intended for the primary need of the client user (Members / Consumers). Members / Consumers are the users across the country that will access the viewing of the product information if the market in their place follows the suggested retail price of the commodities and if not they will send complaints according to their actual experiences. Members / Consumers have also the right to access the Forum with different rights if you are already registered to the system you might be able to join the asynchronous conversion with the different members across the country and if not yet registered the consumers will still view the minimal conversation of the members.

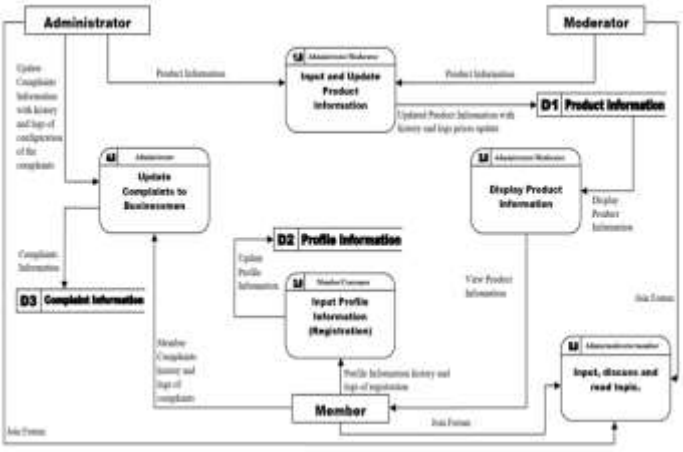


Figure 2. Logical Data Flow Diagram

Figure 2 shows the more detailed access rights of different users of the system. The logical Data Flow Diagram discusses the most intensive transaction inside the system developed. It defines specifically the user access right with the different modules of the system. The modules are numbered from 1.0 to 5.0 labeled with users who have the access right. The figure is also labeled if the user is intended for viewing only and if the user is allowed to alter the information.

Figure 3. Hierarchical Input Process Output (H-I-P-O)

The H-I-P-O Hierarchy is being used as an aid during general system design and the output of the general program design is the input for the detailed module design (Stevens, W. P., Myers, G. J., & Constantine, L. L., 1974). The considerations and techniques presented here are useful for evaluating alternatives for those portions of the system that will be programmed on a computer.

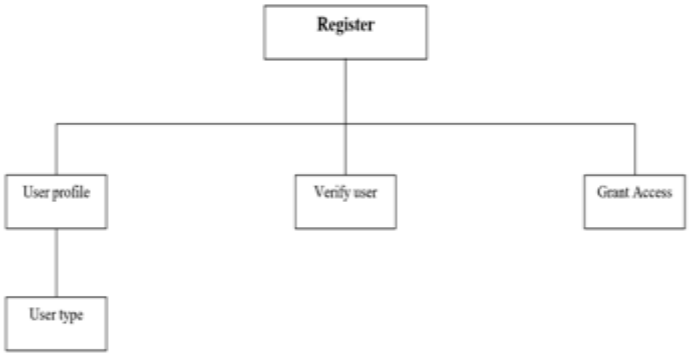
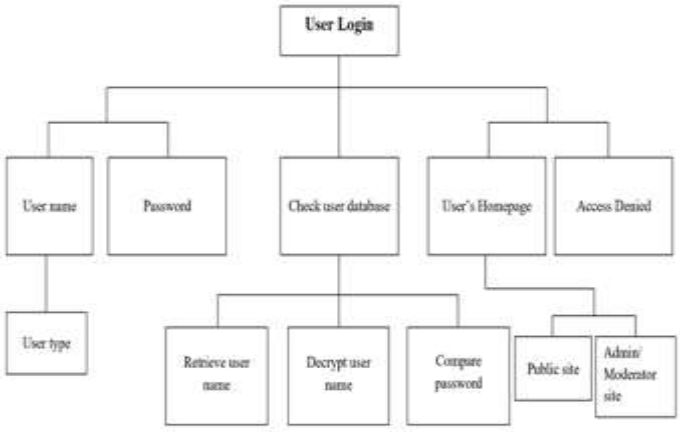


Figure 3.1. H-I-P-O Diagram for Register

In the H-I-P-O Diagram for Register, starting in identifying the user type as shown in Figure 1 there are three rectangles composed of the Administrator, Moderator, and Members / Consumers. It is important to give identity to each user to ensure the accountability of each user. The user profile will be provided by the user who registers, then verified, and then grant access.



3.2. H-I-P-O Diagram for Access Control

The H-I-P-O Diagram for Access Control is designed to verify the user type and password as the security to avoid malicious user access. This will also identify the limitation of user access in every module provided by the system.

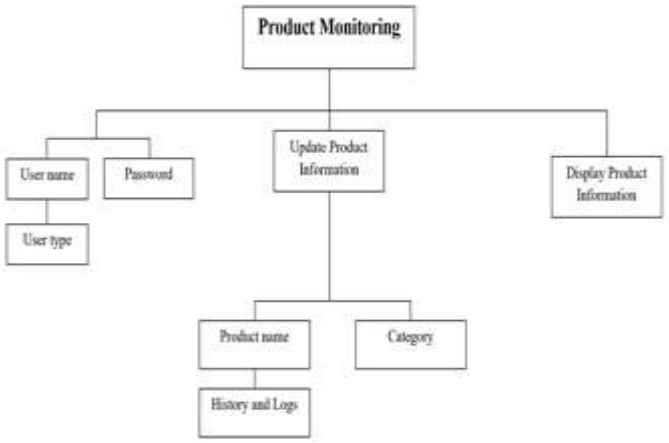


Figure 3.3. H-I-P-O Diagram for Product Monitoring

The H-I-P-O Diagram for Product Monitoring is the main display of the developed system, and as observed user type is also identified to avoid discrepancies in the display of product information. This module is the same as the administrator and moderator changing the product information so that the user type is very important to include.

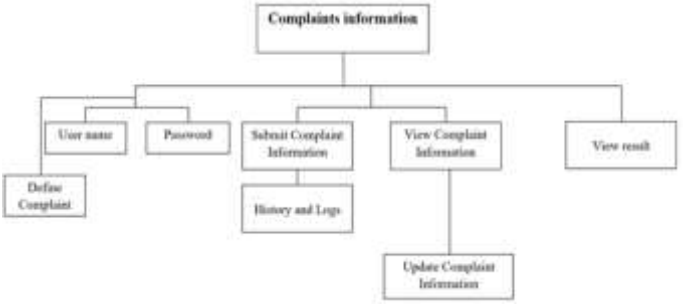


Figure 3.4. H-I-P-O Diagram for Complaint Information

The H-I-P-O Diagram for Complaint Information, this module shows the data flow of the complaints submitted by the members and consumers.

III. RESULTS AND DISCUSSIONS

Table 1. Data Dictionary

Table 1.1. Answers

Name	Type	Length	Decimals	Allow Null	
commentid	int	11	0	☐	🔑1
author	varchar	250	0	☐	
forumId	int	11	0	☐	
answer	varchar	10000	0	☐	
datecreated	datetime	0	0	☐	

Table 1.2. Commodity Type

Name	Type	Length	Decimals	Allow Null
comId	int	11	0	
description	varchar	256	0	
class	varchar	5	0	

Table 1.3. Complaint Response

Name	Type	Length	Decimals	Allow Null
complainresponseid	int	11	0	
complainid	int	11	0	
message	varchar	1000	0	
from	varchar	100	0	
datecreated	datetime	0	0	

Table 1.4. Complaints

Name	Type	Length	Decimals	Allow Null
complainid	int	11	0	
fullname	varchar	268	0	
email	varchar	50	0	
respondent	varchar	255	0	
respondentaddress	varchar	256	0	
description	varchar	10000	0	
datecreated	datetime	0	0	

Table 1.5. Current Prices

Name	Type	Length	Decimals	Allow Null
cur_priceid	int	11	0	
productId	int	11	0	
low	float	10	2	
high	float	10	2	
prevailing	float	10	2	
location	varchar	256	0	
remarks	varchar	1000	0	
datecreated	datetime	0	0	
dateupdated	datetime	0	0	☑
updatedby	varchar	50	0	☑

Table 1.6. Forum

Name	Type	Length	Decimals	Allow Null
ForumId	int	11	0	
subject	varchar	1000	0	☑
Author	int	11	0	☑
Description	varchar	1000	0	☑
DateCreated	datetime	0	0	☑

Table 1.7. Members

Name	Type	Length	Decimals	Allow Null
memberid	int	11	0	
title	varchar	255	0	☑
firstname	varchar	50	0	
lastname	varchar	50	0	
email	varchar	50	0	
occupation	varchar	128	0	☑
username	varchar	20	0	
password	varchar	20	0	
datecreated	datetime	0	0	☑
dateupdated	datetime	0	0	☑

Table 1.8. Price Update Logs

Name	Type	Length	Decimals	Allow Null
priceupdatelogid	int	11	0	
productid	int	11	0	
location	varchar	256	0	☑
srp	float	0	0	
datecreated	datetime	0	0	
updatedby	int	11	0	☑

Table 1. Data Dictionary is a design table accordingly to the data flow diagram in the methodology of the research presented. Hatley, D., & Pirbhai, I. (2013), stated that there has been growing dissatisfaction with conventional strategies for systems development, and an increasing number of proposals for replacing them with new, more formal approaches.

Figure 4. Entity Relationship Diagram

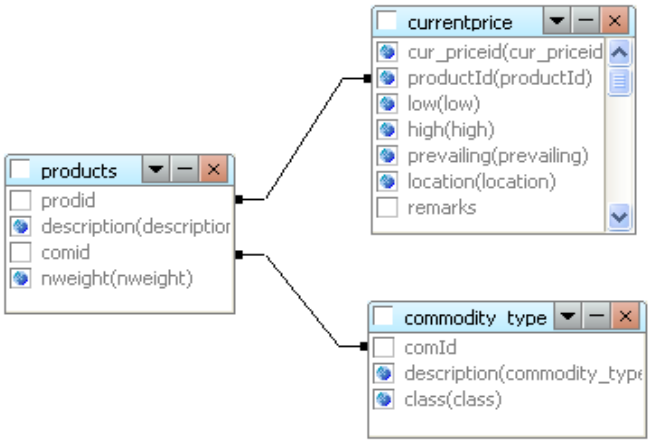


Figure 4.1. products, currentprice and commodity type Relationship (one-to-one)

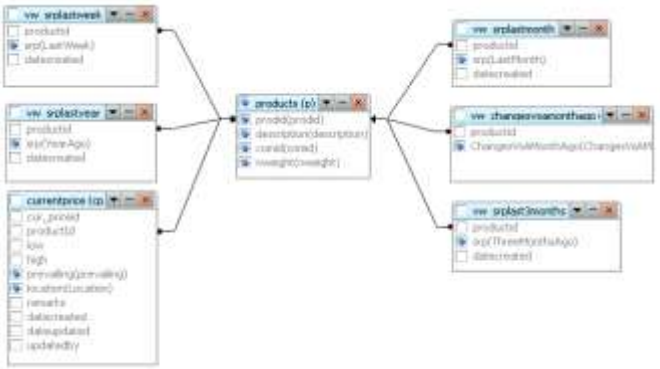


Figure 4.2. products (p) and vw_srplastweek, vw_srplastmonth, vw_srpLast3months, vw_srplastyear and vw_srpchangesvsmonthago views Relationship (one-to-many)

Figure 4. The entity Relationship Diagram is a data model that represents the logical structure of the database (Rossi, B., 2014).

IV. CONCLUSION

This study gets the Data Flow Diagram (DFD) in Developing Online Product Monitoring System of DTI. In developing a system the programmer must create first the data flow of the system, before the development or the system creation will start. As the claim in DFDs theory, the level of data flow diagram could go beyond level 3, but rarely do this. Bangeter, J. (July 06, 2017) stated this, this author anticipates already that the data flow diagram can be pushed through the most intensive definition of the system development and in more specified accordingly and necessarily.

V. RECOMMENDATION

It is strongly recommended by the author of this research to use the same method in developing a system. After reading this research you will have some background knowledge on data flow diagrams and how they are categorized, and you're ready to build your own DFD. Data flow diagrams as applied in the analysis of software development enhanced the security and order of the developed online system.

REFERENCES CITED

[1] Li, H., Sun, X., Shi, P., & Lam, H. K. (2015). Control design of interval type-2 fuzzy systems with actuator fault: Sampled-data control approach. Information Sciences, 302, 1-13. Retrieve August 11, 2018 from https://goo.gl/CsHUQq

[2] Hashem, I. A. T., Yaqoob, I., Anuar, N. B., Mokhtar, S., Gani, A., & Khan, S. U. (2015). The rise of "big data" on cloud computing: Review and open research issues. Information Systems, 47, 98-115. Retrieve August 11, 2018 from https://goo.gl/P2Kpei

[3] Ravichandran, H., Toohil, R., & Lammi, L. (2014). U.S. Patent No. 8,843,571. Washington, DC: U.S. Patent and Trademark Office. Retrieve August 11, 2018 from https://goo.gl/sMgivH

[4] Liu, S. (2013). Formal Engineering for Industrial Software Development: Using the SOFL Method. Springer Science & Business Media. Retrieve August 11, 2018 from https://goo.gl/x6x6tu

[5] Abedjan, Z., Schulze, P., & Naumann, F. (2014, November). DFD: Efficient functional dependency discovery. In Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management (pp. 949-958). ACM. Retrieve August 11, 2018 from https://goo.gl/Sgx2Vx

[6] Stevens, W. P., Myers, G. J., & Constantine, L. L. (1974). Structured design. IBM Systems Journal, 13(2), 115-139. Retrieve August 19, 2018 from https://goo.gl/sMVqtf

[7] Hatley, D., & Pirbhai, I. (2013). Strategies for real-time system specification. Addison-Wesley. Retrieve August 19, 2018 from https://goo.gl/Rbs6B4

[8] Rossi, B. (2014). Entity relationship diagram. Retrieve August 19, 2018 from https://goo.gl/hYRmte

[9] Bangeter, J. (July 06, 2017). Data Flow Diagram Symbols, Types, and Tips. Retrieve August 19, 2018 from https://goo.gl/bHEJi2