

# Development of a Web-Based Convergent Hospital Billing System

*Adhistya* Erna Permanasari\*, *Silmi* Fauziati, *Argi* Kartika Candri

Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada, Jl. Grafika no 2, Yogyakarta, 55281, Indonesia

**Abstract.** Given the large volume of data generated in hospitals, the usage of hospital information system (HIS) that may ease the hospital's workload on managing patient bills is critical. Thus, this paper presents the development of a web based hospital billing system which allows the staffs to manage and store patient bills in such a way that it can integrate bills from various departments. In addition to that, the system has additional features to allow down payments, automatically generate accounting reports, as well as integrate some features of the registration and the pharmacy system. The system design followed the Waterfall approach and was done using the use case diagram, activity diagram, sequence diagram, and entity relationship diagram. The data processed was actual data from the hospital and dummy data for the pharmacy system. The system's functionality was tested using the black-box method and then evaluated by the hospital staffs using the SUS (System Usability Scale) method. From there, we obtained a score of 77.5 out of 100 from the SUS evaluation and 100% success rate from 102 features tested with the black-box testing method. Therefore, we can conclude that the new system works well and has good usability.

## 1 Introduction

As one of the most important social organizations that play a major role in providing health-treatment services to the citizens, hospitals always deal with the large volume of collected data, mainly transactions and bills. Hospitals are no exception to this. Thus, implementing Information Technology (IT) in the form of a Hospital Information System (HIS) is essential to manage patient bills in hospitals efficiently.

Vegoda defined Hospital Information System (HIS) as an information system which is integrated in such a way to improve patient care by providing information necessary to allow an increase in the user's knowledge and reduce in uncertainty, thus allowing rational decisions to be made [1]. In HIS, different software are integrated to grasp the whole data in certain hospital sections. A good HIS should be able to cover various departments and personnel of the hospital. Meanwhile, there are few ongoing trends in the development of hospital billing system to this date [2]. The trends include classifying patient bills based on the services given, integrating bills from the various department, integrating payment module as well as the accounting module.

---

\* Corresponding author: [adhistya@ugm.ac.id](mailto:adhistya@ugm.ac.id)

Another popular trend in hospital billing system development is breaking down the bills to explain where the down payment went forth. In addition to that, Kavatra et al., also mentioned that one of the most important features of a billing system is to support down payment [3]. According to them, down payment should be compulsory and must be paid in advance to ease the hospital's burden when collecting the patient bills. Finally, Lin argued that hospital bills should be broken down into tiniest details and the one charged to the patient should have already been summarized from bills charged by various departments [4].

This paper presents a convergent hospital billing system, developed to help the staffs in calculating and printing patient bills. The term convergent means that we developed the system in such a way that it can integrate invoices from various services into one single invoice. In addition to that, we also integrated features to allow advance payment as well as automatically generated financial reports. Likewise, we also integrated some features of the registration and pharmacy system. We designed the system using waterfall software development method. Later, we asked the staffs of hospital 'X' where this study took place, to test the system usability using the System Usability Scale (SUS) method. Moreover, we also conducted a test on system functionality using the black-box method. The data processed in the conducted tests was actual patient data and dummy data for the pharmacy system.

The remainder of the paper is structured as follows. Section 2 describes methods. Section 3 presents system architecture. Section 4 reports results and analysis. Finally, Section 5 presents the conclusion of the study.

## **2 Methods**

We developed the system on PCs using HTML, CSS, JS, and PHP programming language. The workflow processed in this system was adapted from the actual system flow running in Hospital 'X'. Meanwhile, we adapted the waterfall software development model in this research for its simplicity, flexibility and straightforward process. Those stages include requirement analysis and definition; system and software design; implementation and unit testing; and at last, integration and system testing [5].

### **2.1 Requirement analysis and definition**

In this step, we gathered and analyzed requirements needed to build the system. The requirements were collected from various sources and a series of meetings with the hospital staffs. Throughout the gathering process, we sought to understand how the end-user hoped to use the system. Later, we successfully identified the actors and their use cases.

### **2.2 System and software design**

After the requirements gathering was done, we designed some diagrams to understand the system workflow better. The diagrams are the use case diagram, activity diagrams, and sequence diagrams. The use case diagram was used to model the interactions between users and the tasks performed by the system. It is illustrated in Figure 1.



**Fig. 1.** Use case diagram of the system.

Here, we designed the front desk staff as the main actor, since they are the ones responsible for dealing with customers and document their bills. Later, the use case diagram was further described using interaction diagrams, which are the activity and sequence diagrams. After finished with the diagrams, we also designed low-fidelity prototypes to understand better the system workflow and how it should have worked.

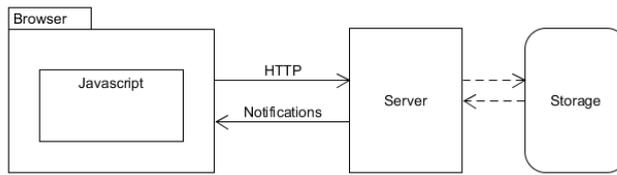
### 2.3 Implementation and unit testing

In the last step, we implemented the features into one working system and tested it to ensure its reliability. The system's functionality was tested using the black-box method and then evaluated by the hospital staffs using the SUS (System Usability Scale) method.

## 3 System architecture

In this research, the system was implemented as a web application and accessible using a browser. The system architecture can be seen in Figure 2. Since it was still initial design, the system architecture is quite a simple one. The database was built in a local server and could be accessed through the HTTP protocol. The login rights assignment has yet to be implemented since there are still many changes to come, and thus implementing it at an early stage will probably disrupt the workings of the whole system.

Here, the data was stored in a local database, and the administrator could access the data with an internet connection. We developed the system by using the CodeIgniter framework for its easy configuration and flexibility of going into the very depth of development. In addition to that, CodeIgniter framework offers the easiest way to utilize modular programs and get the specific function through the framework.

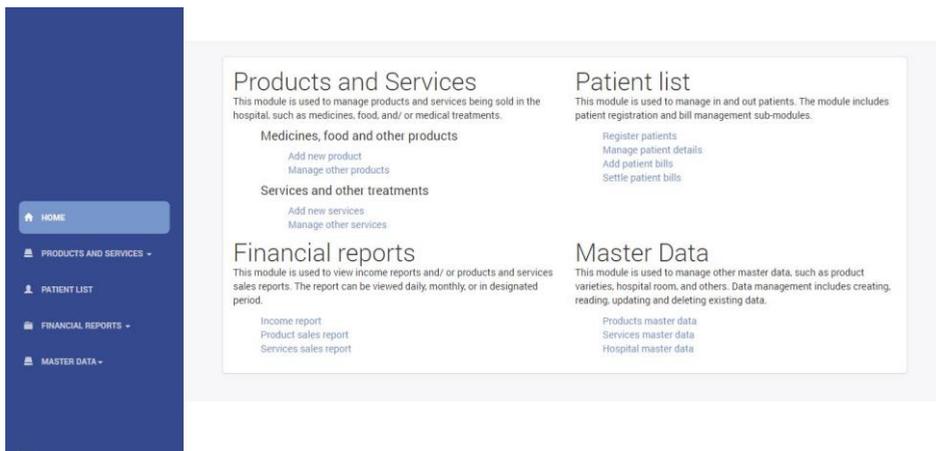


**Fig. 2.** System architecture.

## 4 Results

### 4.1 System implementation

By default, the user will be greeted with the home page when they started the system. The home page contains a handy collection of quick links to the features deemed most important to the user. In total, there are five options available on the menu, which are Home, Products and Services, Patient Directory, Financial Statements, and Master Data. Just as the name suggested, we could access the features to manage patients by clicking on the Patient Directory menu. We might also view the sales reports and financial statements by clicking on the Financial Statements menu. Figure 3 presents the homepage of the billing system.



**Fig. 3.** Billing system interface.

### 4.2 System evaluation

In this research, evaluation is done by using three methods: System Usability Scale (SUS) test, black box test, and implementation test at Hospital 'X'. SUS functions to test application usability, black box test is used to test whether the application is working properly or not, and the implementation test is intended to find out whether the application is ready to be implemented or not.

#### - Black box

Black box testing is a test used to evaluate the fundamental aspects of the application is the application can work well or not without regard to the internal logic structure of the application. Testing is done by testing each function in the application. Based on the test results, the system functionality of the application can work as expected.

- System Usability Scale (SUS) test

We conducted Testing System Usability Scaling (SUS) with respondents from the Hospital 'X'. One of SUS advantage is that it can be used on small sample sizes with reliable results [6,7]. The respondents were three doctors. Selection of respondents tailored to the usefulness of the application is the application used to record patient medical record data in Hospital 'X'. Respondents are taken based on their direct association with the application and represent each related profession. The test results are shown in Table 1.

**Table 1.** SUS Evaluation Results.

	Old system			New system		
	U1	U2	U3	U1	U2	U3
Total	30	18	27	35	28	30
SUS Score (Total x 2.5)	75	45	67.5	87.5	70	75
Average Score	62.5			77.5		
Usability Value Improvement	$\frac{77.5 - 62.5}{62.5} \times 100\% = 24\%$					

SUS assessment is achieved by summing the scores for each question. Each point of the question contributes a value of 0 to 4. The value of odd number question (1, 3, 5, 7, and 9) is the position of the Likert value minus 1. As for the value of the even number (2, 4, 6, 8, and 10) is five minus Likert scale value. After the addition of each point value, the total value is multiplied by 2.5 to get the SUS value. The results of the system reusability test in Table 2 show that the user recognizes that the new system usability value is better than the old system ( $77.5 > 62.5$ ) and the new system usability value exceeds the acceptable usability limit value ( $77.5 > 68$ ). It can be seen that there is an increase in usability value of 24% compared to the previous system. Meanwhile, the functionality test results show a 100% success rate which means the system is running well.

## 5 Conclusions

This study presents a system of patient billing calculations and financial reports. The patient's bill is calculated based on the amount of product/drug bills and actions charged to the patient during the patient's treatment at the hospital. Financial statements are automatically generated based on the related input. Test results showed an increase of 24% compared to the old system. It indicates that the user well receives the proposed system.

## References

1. P.R. Vegoda, *Int. J. Clin. Monit. Comput.*, **4**, 2 (1987).
2. P. Kumar and K. Kosalram, *Int. J. Inf. Eng. Electron. Bus.*, **1**, (2013).
3. S. Kavatra, A.N. Nagappa, P. Kumar, and R.K. Singla, *Indo Glob. J. Pharm. Sci.*, **3**, 3 (2013).
4. J. Lin, "Trends in Healthcare Payments and Billing", (2015).
5. I. Sommerville, "Software Engineering", 9th ed., Pearson Education, (2011).
6. J. Brooke, *Usability Eval. Ind.*, **189**, 194 (1995).
7. A. Bangor, P. Kortum, and J. Miller, *J. Usability Stud.*, **4**, 3 (2009).