MEDINFO 2021: One World, One Health – Global Partnership for Digital Innovation
P. Otero et al. (Eds.)

doi:10.3233/SHTI220051

Design and Implementation of Model-Driven Development for Nursing Information System

Pei-Ying Wen, Shien-Young Chang

Institute of Medical Sciences, Tzu chi University, Hualien, Taiwan

Abstract

Current healthcare industries are more focusing on quality, efficiency, and safety. Nursing information systems are among the key components of healthcare information systems; they can help nurses to use data more effectively while managing their nursing activities, allowing them to improve patient care and develop their knowledge. According to clinical various needs and various hospitals, so as to develop a flexible Nursing Information System, this study based on the MVC, Object-Oriented Programming is used to develop the web interface, and web service is used to exchange data with HIS. This design and implements a flexible Nursing Information System which is applied and summarized clinically on the function of acquiring clinical data from the different clinical requirements and showing the web pages and different kinds of forms from the device browser. Under the software development environment of high variability, high complexity, and high customization in the medical industry, a scalable and maintainable nursing information system is realized through the MVC architecture and the Master-Detail data table design, which can meet the needs of different organizations.

Keywords:

Model-View-Controller Nursing Information System Electronic Medical Record

Introduction

Technology is changing the world at an astonishing speed, driving innovations in various industries, and has also unsurprisingly changed the medical care industry. The development of smart hospitals has become one of the important development strategies for medical institutions around the globe in the future. Smart hospitals rely on the sharing and utilization of information. In order to meet the needs of various cloud-based and mobile information applications, a scalable and maintainable information system architecture is of vital importance. In the Hospital Information System (HIS), the Nursing Information System (NIS) is the most critical component [1]. It helps nurses use data more effectively while managing clinical nursing activities so that they can improve the quality of patient care and develop their knowledge [2]. In addition to a patient-centered design, NIS must have the scalability to expand according to the needs of different clinical users. A hierarchical architecture should be used to add system functions, and a modular, parametric, and component-based design should be used to for maintenance and expansion, so as to improve the clinical care quality and work efficiency with innovative technologies. the system architecture of NIS must be able to meet the requirements of diversified hardware devices, support any devices such as desktop computers, tablets, and mobile phones, and be applicable to all operating systems such as Windows, Android, and iOS. Furthermore, after unified update in the background, all clinical users can update the system functions and contents synchronously [4]. In summary, the main purpose of this study is to design and develop a reliable, efficient, and flexible NIS with a nursing process according to various clinical needs and different hospitals, which can be presented on different devices (e.g., computers, mobile devices, etc.) and used in different clinical scenarios.

Methods

In this study, we expect to build a reliable, efficient, and flexible nursing information system through system design and core architecture. We will use the three-tier Model-View-Controller, i.e., the MVC architecture, as shown in the following figure 1. JavaScript and HTML will be combined to develop the User Interface (UI), and Web service will be used to exchange data with the Hospital Information System (HIS). In this program, the Object-Oriented Programming (OOP) design will be adopted to plan the encapsulation and logic programming of all system applications, the related data, and the data processing methods. Another vital core structure is database design, which will adopt the Master-Detail structure to further increase the flexibility in modifying and maintaining the Nursing Information System according to different needs.

Model-View-Controller (MVC) is a system design pattern that aims at realizing a dynamic program design, simplifying the subsequent modification of the program and expansion of functions, and allowing a certain part of the program to be reused. Besides, it also hides the back-end business logic and thus plays a protecting role. The V(View) in MVC is responsible for User Interface (UI) processing, including the program logic for screen display. The C(Controller) is responsible for receiving users' requests to the system, processing users' input, controlling the program flow, and coordinating the response results of Model and View. The M(Model) is responsible for the intermediate program logic, data processing, and database operations.

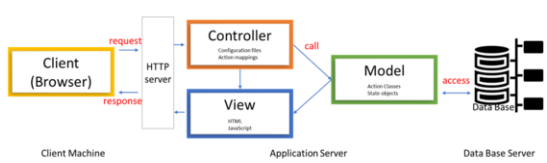

Figure 1– Three tier MVC architecture

The Object-Oriented Programming design takes up a large number of hours in the development and maintenance of many large-scale systems. Usually, the maintenance of systems and programs account for the majority of hours, especially in medical care information systems. Therefore, the readability and scalability of programs are very important, and the concept and writing methods of Object-Oriented Programming (OOP) are the solution to make programs possess these two characteristics. OOP greatly improves the reusability of software. It is mainly composed of a group of abstract classes and concrete classes, in which the reusable interaction modes and control flows between classes are described.

In the database design of NIS, forms are split into a Master-Detail structure. The so-called Master refers to the evaluation date, patient, person making the addition/modification/deletion, and fields (such as the start & end dates and score) that will be used as conditions when searching for a form, all of which are placed in the Master. The form contents, i.e., individual items to be filled in, are all placed in Detail, so that when a form needs to be modified, there is no need to add, delete, or change the fields, and it is also suitable for migration to systems of different hospitals. Therefore, as for the development core, it will be the same database architecture.

Results

NIS is developed and designed with a web-based MVC architecture, which clearly distinguishes the functions of each component and makes it possible to reuse a certain part of the program, thus improving the scalability and usability of the system. It applies the same set of business logic (Model) and the same database contents, and only needs adjustment of different front-end programs (View), so that it can be presented in future systems on different devices (e.g., computers, mobile devices, etc.). Unless there is a service rule change, only the back-end business logic or Controller needs to be adjusted, while the rest of the front-end code can remain unchanged. In addition, it supports the integrated interface of hospital information system data and the automatic acquisition of hospital medical information. By virtue of the web-based MVC architecture, it can be easily integrated with other web-based hospital information systems and provides the convenience of cross-platform usage.

The MVC architecture in NIS is described as follows:

1 The Model has three parts: Model, DBModel, and ViewModel.

1.1 The Model is responsible for logic, data processing, database operation, and data table retrieval. For example, the most commonly used Model is NursingDocumentUtility.cs, which is a method for maintaining the reference numbers of all functions in NIS (such as intake and output) brought into the progress note, as well as adding, linking, and deleting progress notes. If NIS adds new functions and needs to transfer progress notes, used Model is high-risk assessment AssessmentRiskScore.cs, where all the risk levels, names, score ranges, whether a risk is present, whether it is a high risk, color warnings, and front-end text descriptions of risk levels are set for all high-risk assessments. In addition, it includes obtaining the risk definition for specific assessments, and also states the method for inputting the score and obtaining the risk information.

1.2 The DBModel corresponds to the classes in the database, with attributes and field types exactly the same as the structure of the database. The classes in DBModel can be inherited to add, modify, delete, and query data in the database.

1.3 Classes used at the front end of the ViewModel (e.g., the JSON structure required by bootstrap table), Structured Query Language, field structures of SQL strings in multiple tables, and the inherited DBModel classes. The ViewModel files related to this function are all placed in a folder, named by the function name ViewModel, and printed with the function name PrintViewModel, as shown in the following figure 2.

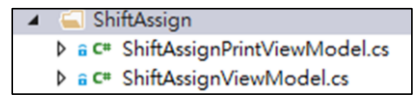

Figure 2– ViewModel

2 The View (front end) is responsible for User Interface, including the display logic. It is divided into ordinary View and PartialView. PartialView: embedded in a certain View.
For example, many functions in NIS are divided into different schools, e.g., for the View of Fall risk assessment, there will be PartialView of multiple schools, such as STRATIFY, Morse, and Hendrich II, as shown in the following figure 3. Pressure Ulcer Assessment includes Braden, Norton, Waterlow, or Admission assessment with classification by age and ward (such as Adult, Pediatric, Obstetrics, Newborn, Intensive Care Unit), etc. The system can easily present the front-end screen according to different hospitals or wards of different natures.

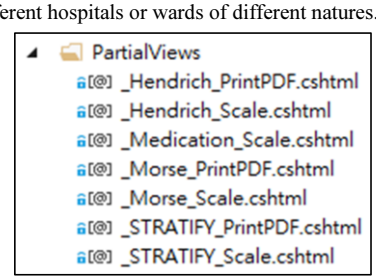

Figure 3– The View of Fall risk assessment

3 The Controller is responsible for receiving requests, processing user input, controlling program flow and coordinating response results of Model and View. In NIS, the most commonly used Controller is the View that transfers data to the front end, as shown in the following table 1.

Table 1– The controller is transfer data to the front-end View

Transfer data to View	
Kernel	**Program code**

JSON	Public ActionResult get List(string feeno, string start_date, string end_date)
	{
	Rectum Content (JsonConvert.SerializeObject(JSON data),"application/json";
	}

ViewData	ViewData["Variable name received by the front end"] = Controller Variable name;
ViewBag	ViewBag. Variable name received by the front end = Controller Variable name;

Model*	public ActionResult InfectiousDiseases_PrintPDF(string beginDate = "", string endDate = "")
	{
	var model = new InfectiousDiseasesPrintViewModel();
	return View(model);
	}

*: Usually only used for printing

Database design is very important for the NIS to have enough flexibility and scalability. Among the most commonly used admission nursing assessment items in many hospitals, there will be different special assessment items according to different age groups or different subjects. In the adult part, there will be Adult Admission assessment and Obstetrics Admission assessment, while the Obstetrics Admission assessment will include assessment contents of Prenatal Checkup and Obstetrics history. In the pediatric part, there will be a Pediatric Admission assessment and a Newborn Admission assessment. Admission assessment for pediatric patients will include the evaluation item of prophylactic inoculation, while Admission assessment for newborns will include the evaluation contents of Mode of Delivery, Apgar Score, Appearance, and Rooming-in, as shown in the following figure 4. However, the design framework of data tables will be the same (as shown in the table 2), so that there is no need to change the data tables because of different subjects or different evaluation items. This is the core concept of Master-Detail, which is also beneficial to program development.

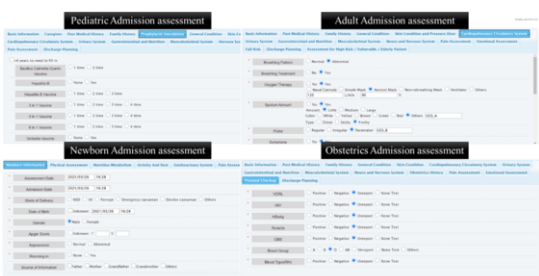

Figure 4– The different admission assessment (Adult Obstetrics pediatric Newborn)

Table 2– The table schema of the Admission assessment

Admission Assessmen – Master

Table Name		Description	
NIS_ASSESSMENT_MASTER		ASSESSMENT_MASTER	
Column name	Data type	Unique key	Comments
FEENO	VARCHA2(20)	FALSE	Patient Number for hospitalized
TABLEID	VARCHAR2(100)	TRUE	Assessment serial number
ASSESS_DATE	DATE	FALSE	Assessment date and time
NATYPE	VARCHAR2(100)	FALSE	Type A:adult、C: pediatric、G: gynecology、O: obstetrics、B:newborn
STATUS	VARCHAR2(10)	FALSE	Status I:insert、A:temporary、U:update、D:delete
CREATE_ID	VARCHAR2(10)	FALSE	Creator ID
CREATE_NAME	VARCHAR2(10)	FALSE	Creator name
CREATE_DATE	DATE	FALSE	Creator date and time
MODIFY_ID	VARCHAR2(10)	FALSE	Editor ID
MODIFY_NAME	VARCHAR2(10)	FALSE	Editor name
MODIFY_DATE	DATE	FALSE	Modify date and time
DELETE_ID	VARCHAR2(10)	FALSE	Deleter ID
DELETE_NAME	VARCHAR2(10)	FALSE	Deleter name
DELETE_DATE	DATE	FALSE	Deleted date and time

Admission Assessmen – Detail

Table Name		Description	
NIS_ASSESSMENT_DTL		ASSESSMENT_DTL	
Column name	Data type	Unique key	Comments
SERIAL	VARCHAR2(20)	TURE	Assessment detail serial number (unique key)
TABLEID	VARCHAR(100)	FALSE	Mapping of Master
ITEMID	VARCHAR(150)	FALSE	Item ID
ITEMVALUE	VARCHAR(2000)	FALSE	Item value
ITEMTYPE	VARCHAR(10)	FALSE	Item type
ITEMID_SN	NUMBER(2)	FALSE	Multiple options under the same Item ID

Conclusions

NIS is a program with a comprehensive care cycle. However, we have actually tried to integrate the daily activities of nurses into the system. Nurses can complete many clinical assessments and complete the work by entering data. For example, fall risk, nutritional assessment, entry and exit records, pipe and drain assessment, etc. After completing all these assessments, NIS can provide clinical alerts based on the user's actual situation. It can notify users and provide some advice messages, such as taking preventive measures or considering other interventions or care. In this sense, it can facilitate decision-making in the daily clinical workflow. In addition, all NIS records and vital signs numbers can be exported to "multidisciplinary progress records" or other required files. This means that it can minimize transcription in daily work and reduce the workload of nursing and the chance of transcription errors. Most importantly, NIS adopts an evidence-based approach and has developed different care plans or pathways so that NIS users can provide protocol-driven care. Many different daily assessments and care treatment plans are provided in the NIS, and they are all based on evidence, such as pain assessment or pressure ulcer risk assessment.

Under the software development environment of high variability, high complexity, and high customization in the medical industry, a scalable and maintainable nursing information system is realized through the MVC architecture and the Master-Detail data table design, which can meet the needs of different organizations. In addition, the characteristics of front-end program planning and design allow users with various devices, such as computers, tablets, mobile phones, and large-screen TVs commonly used in smart wards, to obtain the best visual effects. Since the contents presented are adapted from the system contents, it is not necessary to repeatedly update the system information at great cost. In addition, the input data storage place of any device is the same as the NIS of the computer, so there is no need to perform additional synchronization, which saves the cost of database synchronizing operations and avoids possible data inconsistency errors. NIS has been applied and summarized clinically. It can obtain clinical data from different clinical demands, and display web pages and different forms of functions from the browser on the device.

References

[1] Kahouei M, Molanoroozi Z, Habibiyan M, Sedghi S. Health information technology in the knowledge management of health care organizations. Middle East J Rehabil Health 2015;2(3):e29229.
[2] Rachmawaty R. The development of the electronic nursing record system (ENRS) in the hospital setting: an integrative literature review. Am J Health Sci 2015;6(1):1-6.
[3] Toney-Butler, T.J. and Thayer, J.M. (2018) Nursing, Process. https://www.ncbi.nlm.nih.gov/books/NBK499937/
[4] Bautista, J. R. Filipino Nurses' Use of Smartphones in Clinical Settings. Computers Informatics Nursing, 2019;37(2), 80-89.

Address for correspondence

Pei-Ying Wen is with Institute of Medical Sciences, Tzu chi University, (Phone: 886-970333595; fax:886-3-8579409, e-mail: lavendergin@gmail.com)

Shien-Young Chang, Ph.D. is with Institute of Medical Sciences, Tzu chi University, (Phone: 886-3-8565301; fax:886-3-8579409, e-mail:sychang@mail.tcu.edu.tw)