YASSER B. MOHAMED ARIFIN
WEK020293

RODINA AHMAD

REPORT GENERATION TOOL

ACKNOWLEDGEMENTS

Upon the end of the internship, I would like to take this opportunity to express my gratitude. Firstly I would like to thank to my supervisor, Ms Rodina Ahmad, who give me advices and guided me in finishing my report. The sharing of experience benefits me a lot and I did learned from the good quality of her.

Secondly, I would like to thank my colleague and my group member who give opinions, ideas and suggestion about what to do in my system and they helping me about what I don't know during finishing my system report.

Last but not least, I would like to express my gratitude to all whom helping me. The co-operation from all parties had made this internship program a success.

ABSTRACT

This proposal, prepared for users who want to get a report like chart, save and print the report using online web based system without install any software. This tool can get data from user and process it to generate report or user can manage data by selected what data they need or categories data by group and choose type of chart. This tool can help user to develop report base on web application. User can select type of analysis and type of chart with 3D or 2D, colorful and animation type. User can choose to make analysis from data such as maximum or minimum, average, min data, medium data, and mode data.

TABLE OF CONTENTS

List of Figures

List of Tables

CHAPTER 1

Introduction

1. Introduction

Report Generation Tool can be used to quickly and effectively create and deploy reports without having the developer to platform program, processing and formatting logic. This online tool include or combine various features (e.g: listing, charts,) and characteristics (navigation, hyperlinks, interactive viewing). It uses PDF files for printing documents or save. The conceptual architecture is provided below:

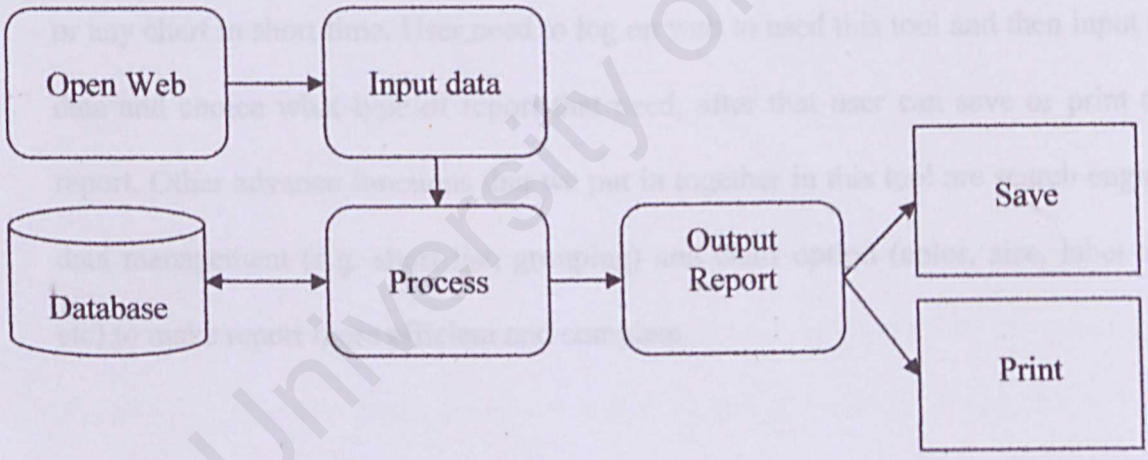


Figure 1.0: Report generation Tool Conceptual Architecture

CHAPTER 2

Project Proposal

1.1 Objective

The objectives of the project:

1. To develop a system that provides the ability to generate report.
2. User can generate report easy and faster by using online application.

Target audience for this tool is anyone that need to analysis data and generate pie, bar or any chart in short time. User need to log on web to used this tool and then input all data and choice what type of report that need, after that user can save or print the report. Other advance functions that we put in together in this tool are search engine, data management (e.g. short list, grouping) and chart option (color, size, label and etc) to make report more efficient and complete.

1.2 Literature Review

The usual of obtaining required report consists of the following steps that making a decision which:

1. information is required;
2. developing a report description;
3. Building of an analytical tables, if necessary;
4. Generating a sequence of SQL queries;
5. Summarizing the obtained data at the client side;
6. Composing the report based on the results and according to the report layout and format requirements;
7. In conclusion, based on the obtained report analysis and decision based on the obtained report analysis and
8. Decision what data are required next.

Some report generator is usually provided in most relational database systems. Such a report generator usually allows the user to specify both the layout and the contents of a report [2]. In some cases, the report layout can be defined via graphical interfaces. Whereas, the majority of the report contents is usually specified using database queries. The benefits of this process are reports can be obtained immediately after inspection data is entered, reports are available to all users in the company, the information is summarized in a way showing what work needs to be completed, and the reports can be modified by the users. The new process takes less time than the

previous method of reading and sorting the paper inspection records. This program is an inexpensive tool that can be used to improve customer service. The reports can be created using simple queries designed from a database program [5]. The reports can be created by the users with some training on database programming.

1.3 Problem Statement

There are a few problems when we want to generate a report:

1) User need to get more fast analysis for their data by online system.
2) Report must include verity of chart type.
3) Difficult to decide what an attribute for data like type of column and row.
4) We didn't know what type of report that user wants.
5) The report only can save as a PDF file but not in other format file like Microsoft Access or Microsoft Word. User can't edit data, chart or file after save because this tool can generate only.
6) User must input all data one by one, no upload file function that content all data like Microsoft® Access or Excel.

1.4 Proposed Solution

We extend to develop the systems that well enable the user to easily produce report. Problems in this tool are come from the data that input from user. If we know what type of data, then we can decide what type of chart that able to be applied and how we present the data in simple way and clear. These entire problems are related between one and other. When we know the first problem then we can finish the next problem. So, the solutions for this problem are step by step instruction to guide user when input the data, before that we must check only a few type of data that can insert in database before go to next step (figure 1). Last step, user will give option to edit and change data before save in PDF format, this step to make sure input and output correctly. Some instruction can help user to understand this tool and help function should include too. Main tools to develop report and user interface are Macromedia ColdFusion MX 7 and Dreamweaver MX 2004. Web language for this system is Dynamic ColdFusion language including rich Flash forms generation, structured reports, and dynamic generation of printable documents. For the database, we use open source MySQL database because more compatible and easy to manage then other database. There are a few limitations for this report generation tool, it can process only one report at the same time that mean user only can get one chart from data and must come back to the first step to make another chart using the same data.

1.5 Methodology

In this project, I selected Rapid Prototyping Methodology to develop this tool. A software rapid prototype is a dynamic visual model providing a communication tool for customer and developer that is far more effective than either narrative prose or static visual models for portraying functionality. Rapid prototyping is basically an analysis technique to discover the complete set of functional requirements for a proposed system. Furthermore, some of these risks can occur disagreements with users and customers regarding methodology, standards, tools and so on, out-of-control users who want to iterate and evolve a prototype into a system that does everything for everyone all of the time, and premature delivery of a prototype instead of a final (thoroughly documented and tuned) product. First, we have done a survey to collect information or feedback from potential users. In this survey, we can get information on the tools or techniques that user use to make a report. At the same time, we need feedback from users about their frequent by used to online application and search engine for basic or advance searching. Based on the analysis 20 person from multiple background and education, there are a few type of chart that user always need to design a report include pie chart (40%), bar/column chart (40%), and line chart (20%). For search engine, user only applies word (62%), sentences (31%), and keyword (25%) for searching method. All candidates decide to mark as user that ever using others tool and search engine (Google and Yahoo Search) for this purpose and always choice to need a formal report (company report or important document). Second,

we need to design a storyboard to describe every step and function that is included in this tool. This storyboard can explain clearly about entire program especially a function that include to make sure no overlapping input from user. After that, interface and prototyping version will develop. Testing error are last step in develop this tool and it will repeat back prototype version if any error will detect.

1.6 Project timeline

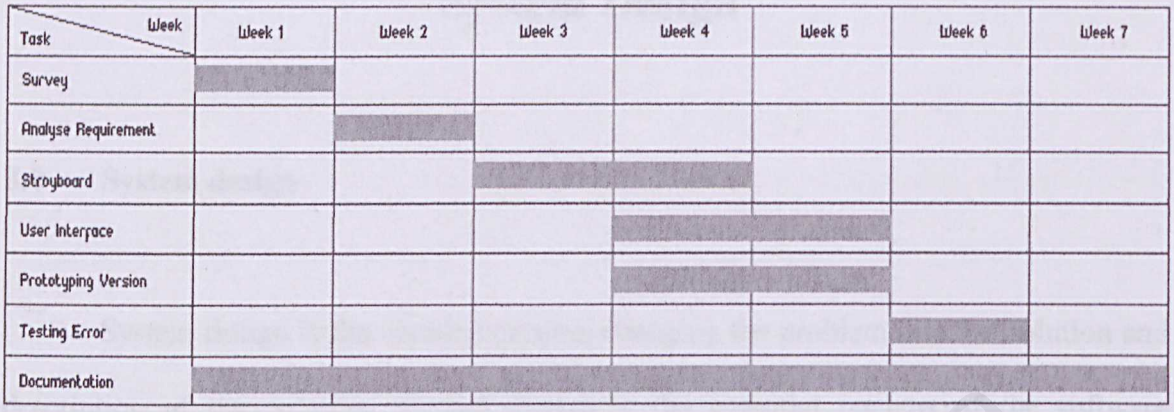


Figure 1.1: Project Timeline for Task in Report Generation Tool

1.7 Summary

Report Generation Tool is one of many ways to develop report but this tool run in World Wide Web or we can call as web application. This simple tool can generate a chart with label and view table of data input. The users are able to print or save the file. Database are use to store the script for built output depend what type of report will user choose. To help user, they can use search engine or data management to manipulate data with sequence determine by user. User only needs to follow four or more step before get the output. To make sure the input are correct, system will check and inform the user to make change. This tool will view chart and report in flash format to make it more interactive and real time data and integrate with ColdFusion language and MySQL database for store syntax.

CHAPTER 3

System Design

3.0 System design

System design is the creative process changing the problem into the solution and description of the solution. System design is the essential nucleus of the software development process and is applied regardless of the development model or standard that is used. The common step involved analyzing, designing, coding and testing the system to ensure that it conforms to the software specification and requirement. Each activity transforms information in a manner that ultimately results in validated computer software.

There can be many ways to implement design, given specific requirements. For example, there many ways to implement graphical user interface design for a website, using image maps, navigational menus or text as hyperlinks and interactive elements. Problems are defined from the requirement given and solutions are considered to solve the problem.

In fact the quality of the system input determines the quality of the system output. It is vital that input form and screens be designed with critical relationship in mind. Well-

designed input screen should meet the objective of effectiveness, accuracy, ease of use, consistency, simplicity and attractiveness.

Effectiveness means that input form and screen serve specific purpose in the management information system, while accuracy refers to design that assures proper completion. Ease of use means that forms and screens are straightforward and require manner that focuses the user's attention. Attractiveness implies that user will enjoy using, or even be drawn to using forms and screens through their appealing design.

Therefore, in the system phase, the system requirement gathered during the system analysis phase and research conducted earlier were translated into a representation of system. Initially, representation that is close to source code. In this phase, input, output, file and database design were produced which include the design of input forms and screens in order to gather input data, data dictionary, file specification and report design.

3.1 Architectural Design

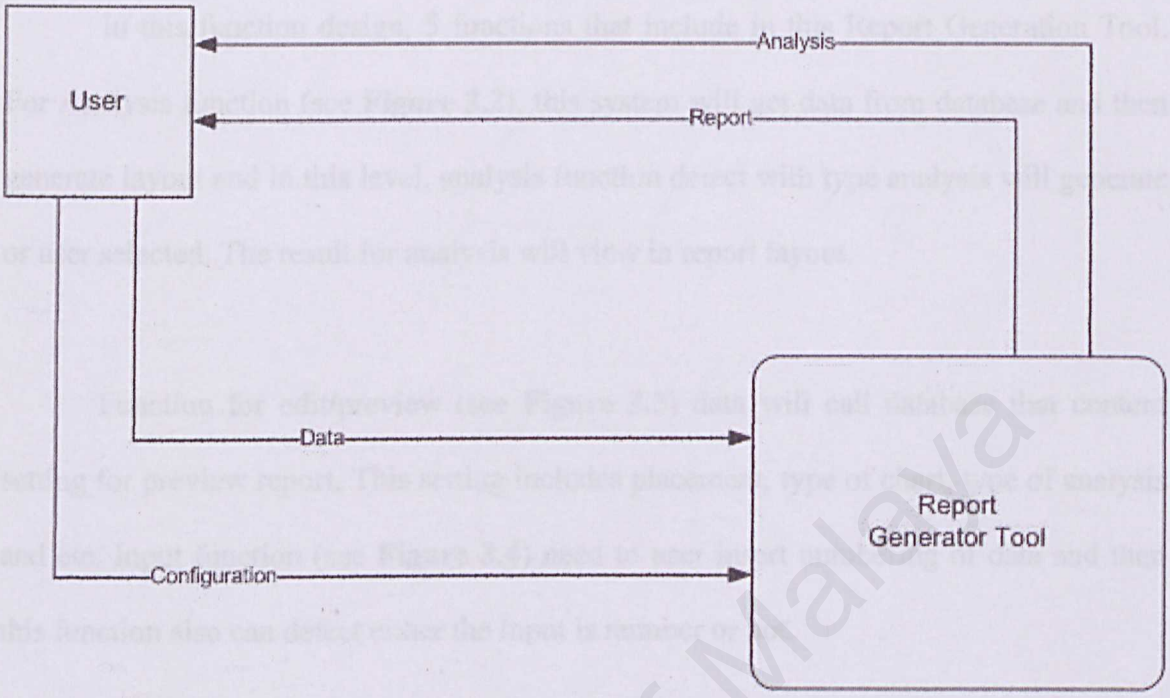


Figure 3.1: Main architectural design for Report Generation Tool

3.2 Function Design

In this function design, 5 functions that include in this Report Generation Tool. For Analysis function (see **Figure 3.2**), this system will get data from database and then generate layout and in this level, analysis function detect with type analysis will generate or user selected. The result for analysis will view in report layout.

Function for edit/preview (see **Figure 3.3**) data will call database that content setting for preview report. This setting includes placement, type of chart, type of analysis and etc. Input function (see **Figure 3.4**) need to user insert numbering of data and then this function also can detect either the input is number or not.

For generate PDF file, function print or save as show in **Figure 3.5** will convert from ColdFusion type to PDF type, so user can select either to print directly or just save the file. The output are depend to setting that user have done in step 5 before this. To design the layout one function Layout will call to generate the interface for report (see **Figure 3.6**).

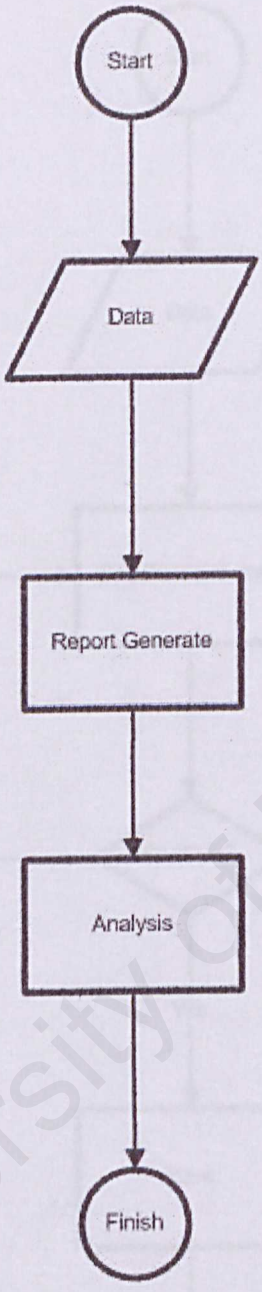


Figure 3.2: Flow Chart for Analysis module

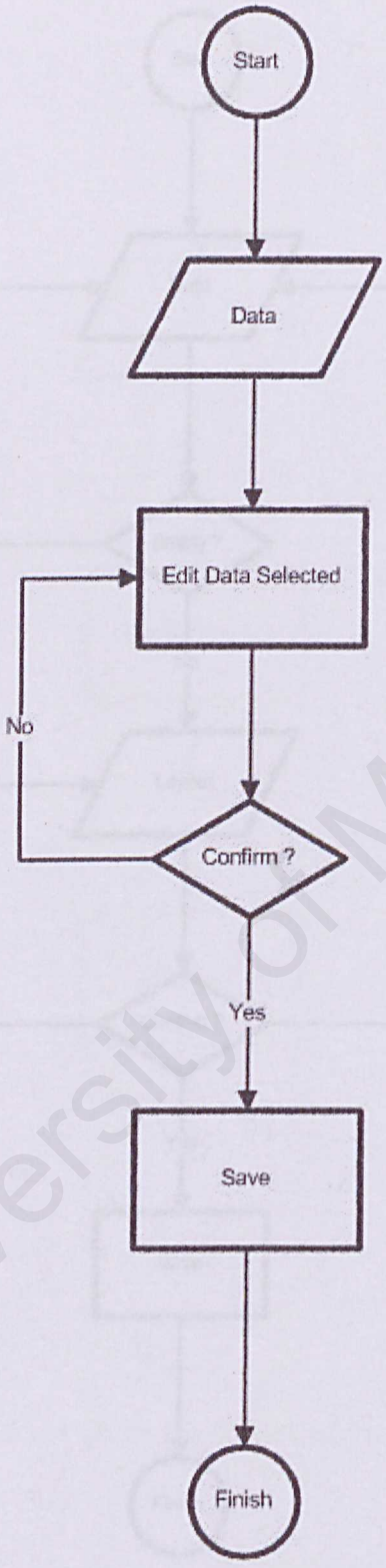


Figure 3.3: Flow chart for Edit Data

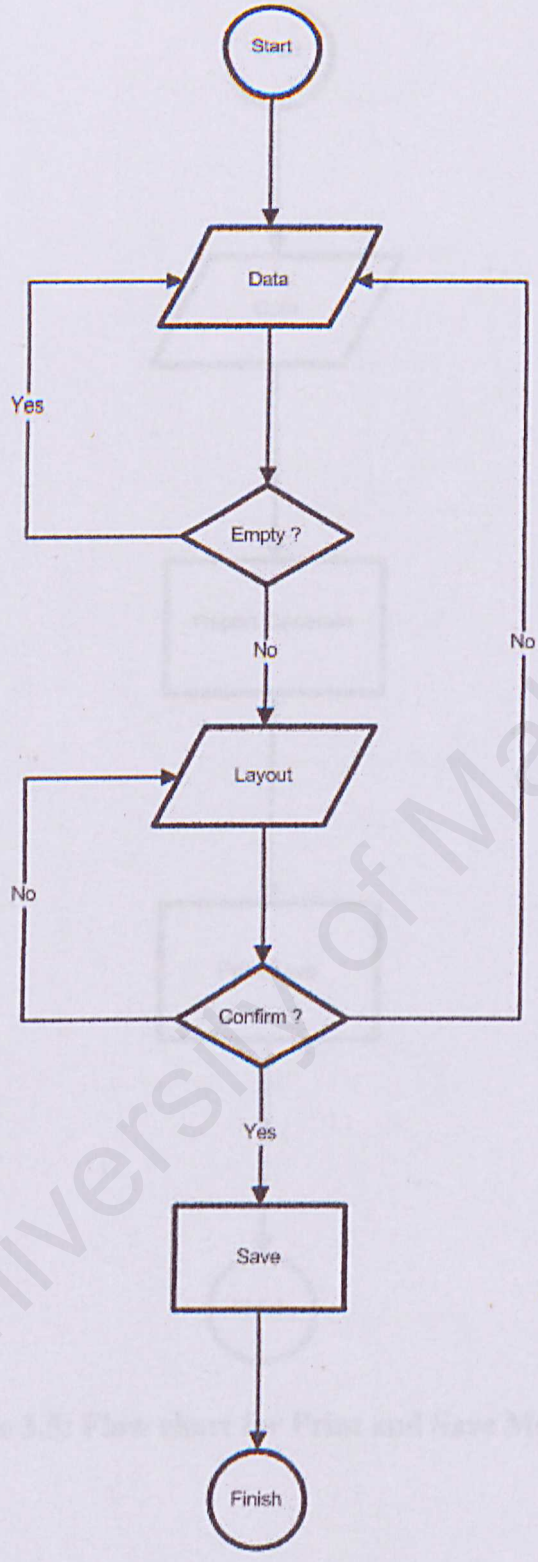


Figure 3.4: Flow Chart for Input Data.

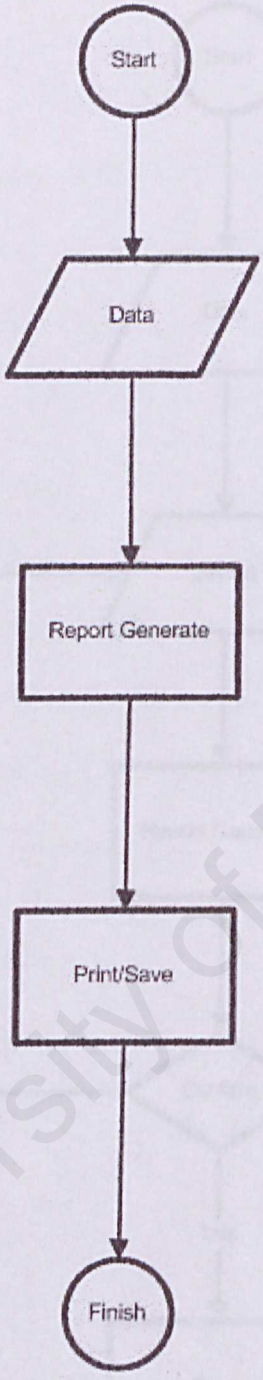


Figure 3.5: Flow chart for Print and Save Module

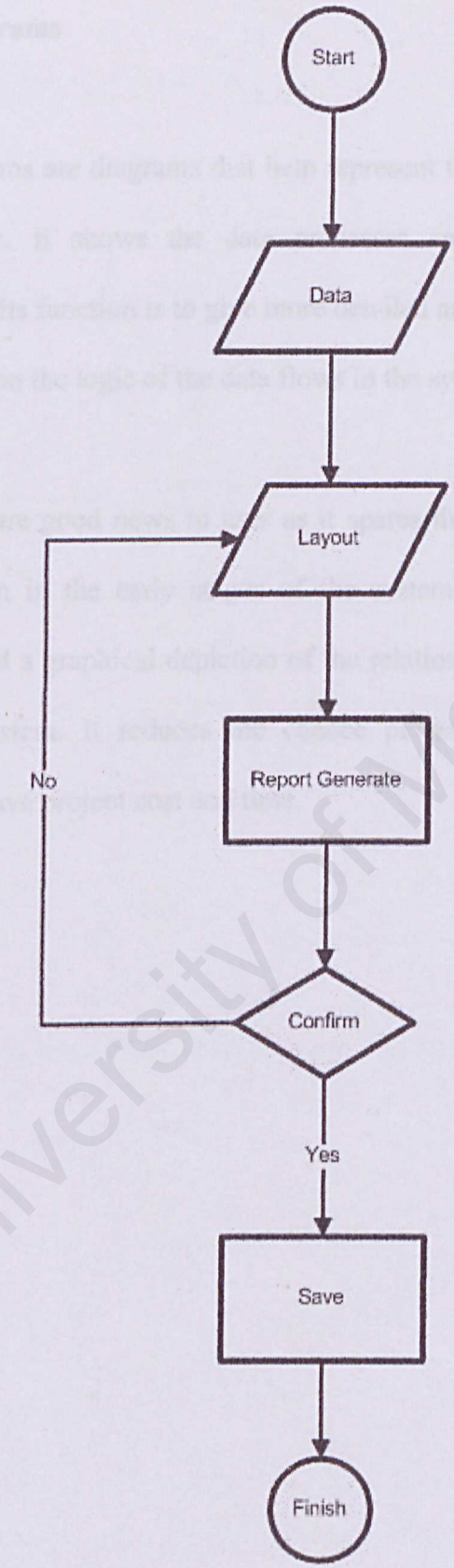


Figure 3.6: Flow Chart for Report Layout/Design

3.3 Data Flow Diagrams

Data flow diagrams are diagrams that help represent the process in the system in more graphical manner. It shows the data processes and flows in an easy and understandable manner. Its function is to give more detailed analysis of the process in the system. It's emphasizes on the logic of the data flows in the system.

These diagrams are good news to user as it spares the user from committing to technical implementation in the early stages of the system development. Its provide deeper understanding and a graphical depiction of the relationship of the each processes and modules of the system. It reduces the chance pf redundancies in the system implementation, which save project cost and time.

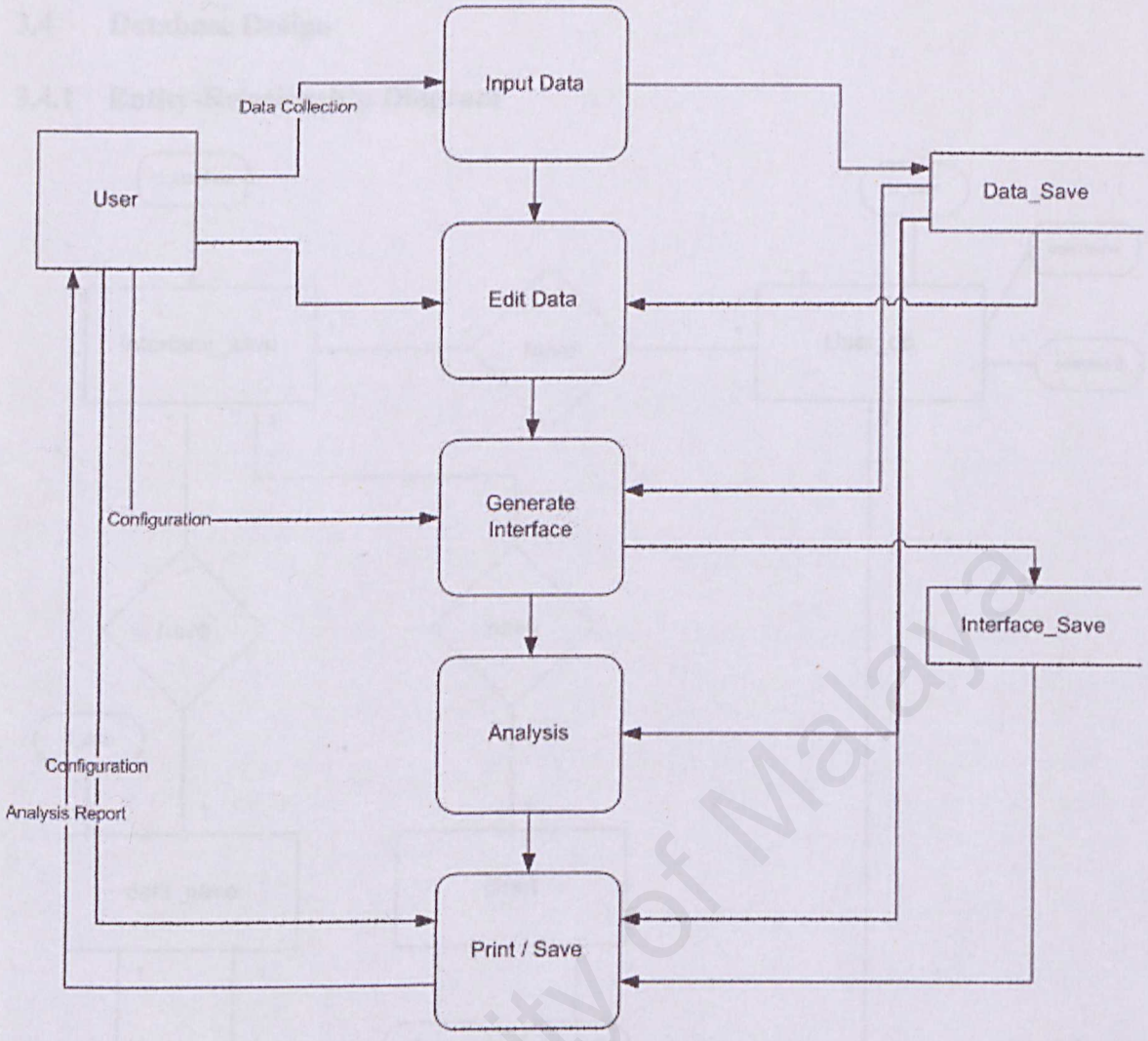


Figure 3.7: Main Data Flow Diagrams for Report Generation Tool

3.4 Database Design

3.4.1 Entity-Relationship Diagram

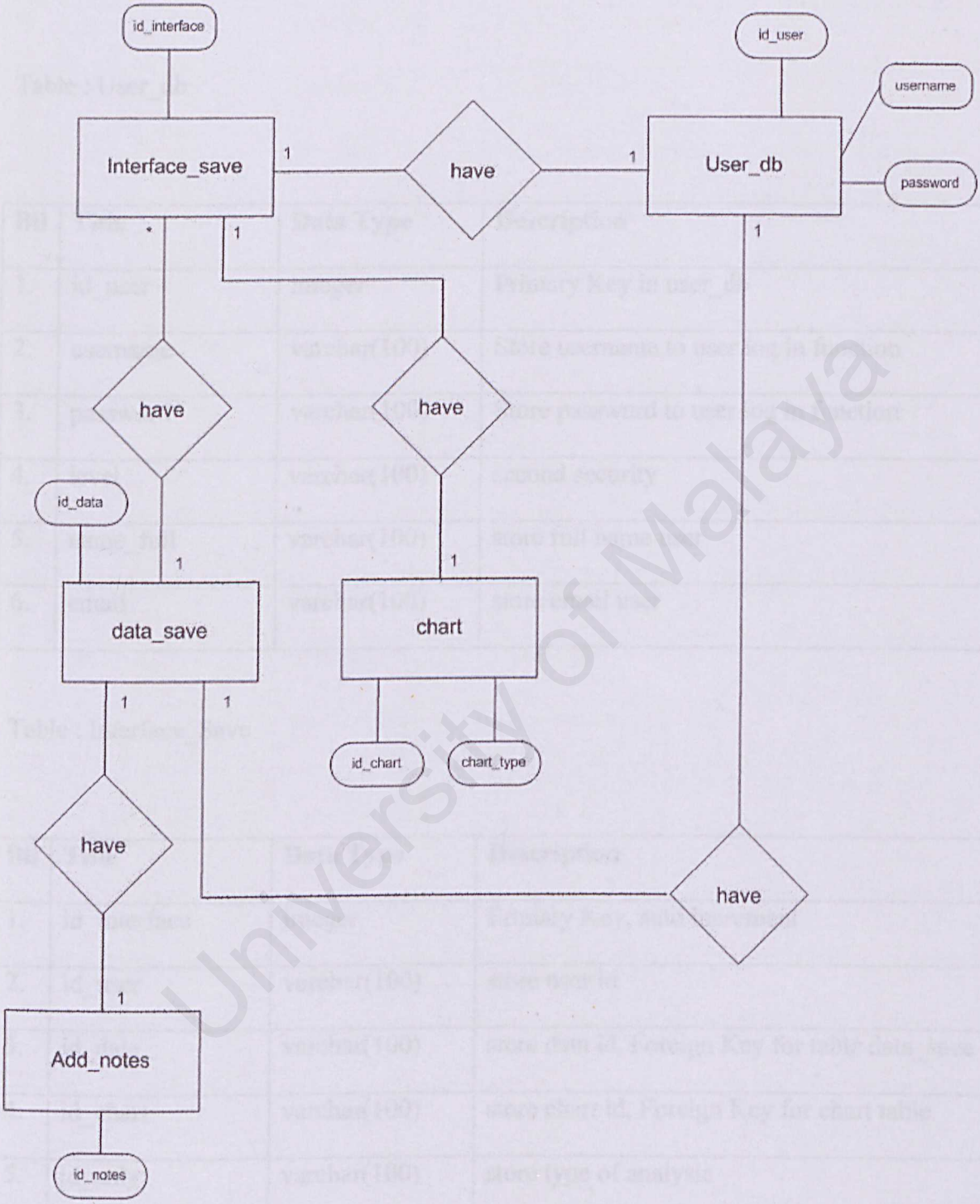


Figure 3.8: Main Entity-Relationship Diagram for Report Generation Tool Database

3.4.2 Data Dictionary

Table : User_db

Bil	Title	Data Type	Description
1.	id_user	integer	Primary Key in user_db
2.	username	varchar(100)	Store username to user log in function
3.	password	varchar(100)	Store password to user log in function
4.	level	varchar(100)	second security
5.	name_full	varchar(100)	store full name user
6.	email	varchar(100)	store email user

Table : Interface_Save

Bil	Title	Data Type	Description
1.	id_interface	integer	Primary Key, auto increment
2.	id_user	varchar(100)	store user id
3.	id_data	varchar(100)	store data id, Foreign Key for table data_save
4.	id_chart	varchar(100)	store chart id, Foreign Key for chart table.
5.	id_anly	varchar(100)	store type of analysis
6.	title	varchar(100)	title of chart

7.	splace	varchar(100)	Layout for chart, 4 type : Default, Cluster, Stacked, and Percent
8.	label_x	varchar(100)	Label for column
9.	label_y	varchar(100)	label for row
10.	legend	varchar(100)	View legend for chart, 2 type : yes or no
11.	show3d	varchar(100)	Show 3D chart, 2 type : yes or no.
12.	int_place	varchar(100)	Layout decoration, 6 type : 1 to 6

Table : Data_Save

Bil	Title	Data Type	Description
1.	id_data	integer	Primary Key, auto increment
2.	date_in	varchar(100)	Date input
3.	id_user	varchar(100)	Store user id
4.	id_view	varchar(100)	List of user id that give permission to see report.
5.	t_row	varchar(100)	Total of row table
6.	t_column	varchar(100)	Total of column table
7.	data_in	longtext	Store all data

Table : Chart

Bil	Title	Data Type	Description

1.	id_chart	integer	Primary Key
2.	title	varchar(100)	Title of chart
3.	chart_type	varchar(100)	Detail about Chart

Table : Add_Notes

Bil	**Title**	**Data Type**	**Description**
1.	id_notes	integer	Primary Key
2.	id_data	varchar(100)	Foreign key in Data Table
3.	date_in	varchar(100)	Date input
4.	title	varchar(100)	Title of notes
5.	notes	longtext	Description about table

Graphical User Interface Design

Input and Output Screen Design

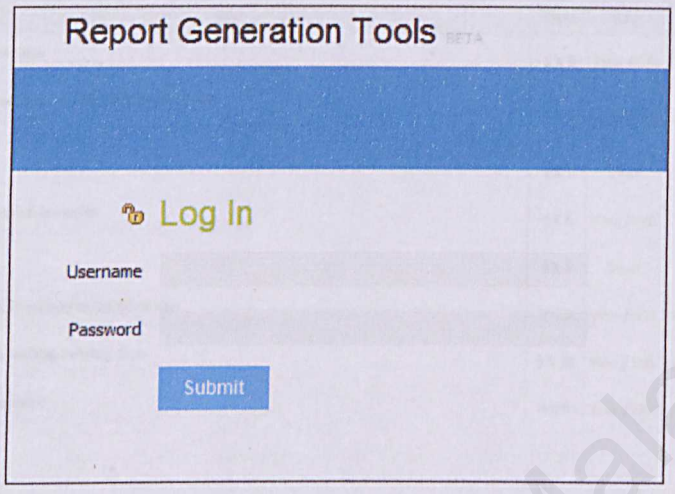


Figure 3.9: Log In to Report Generation Tool

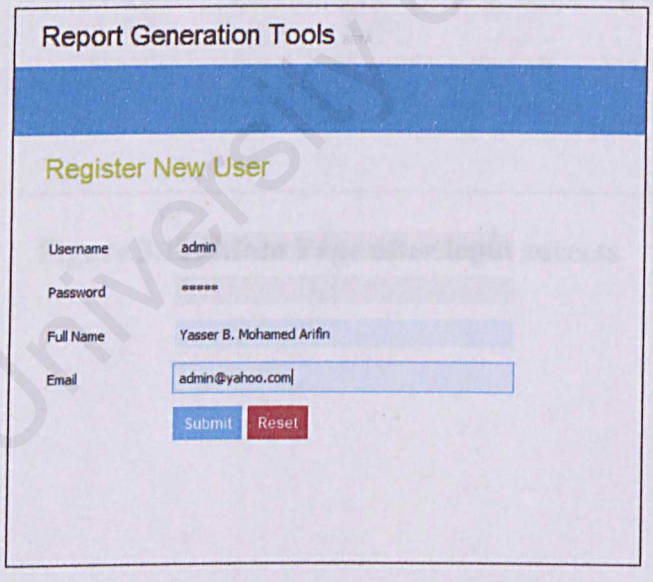


Figure 3.10: Form Register New User

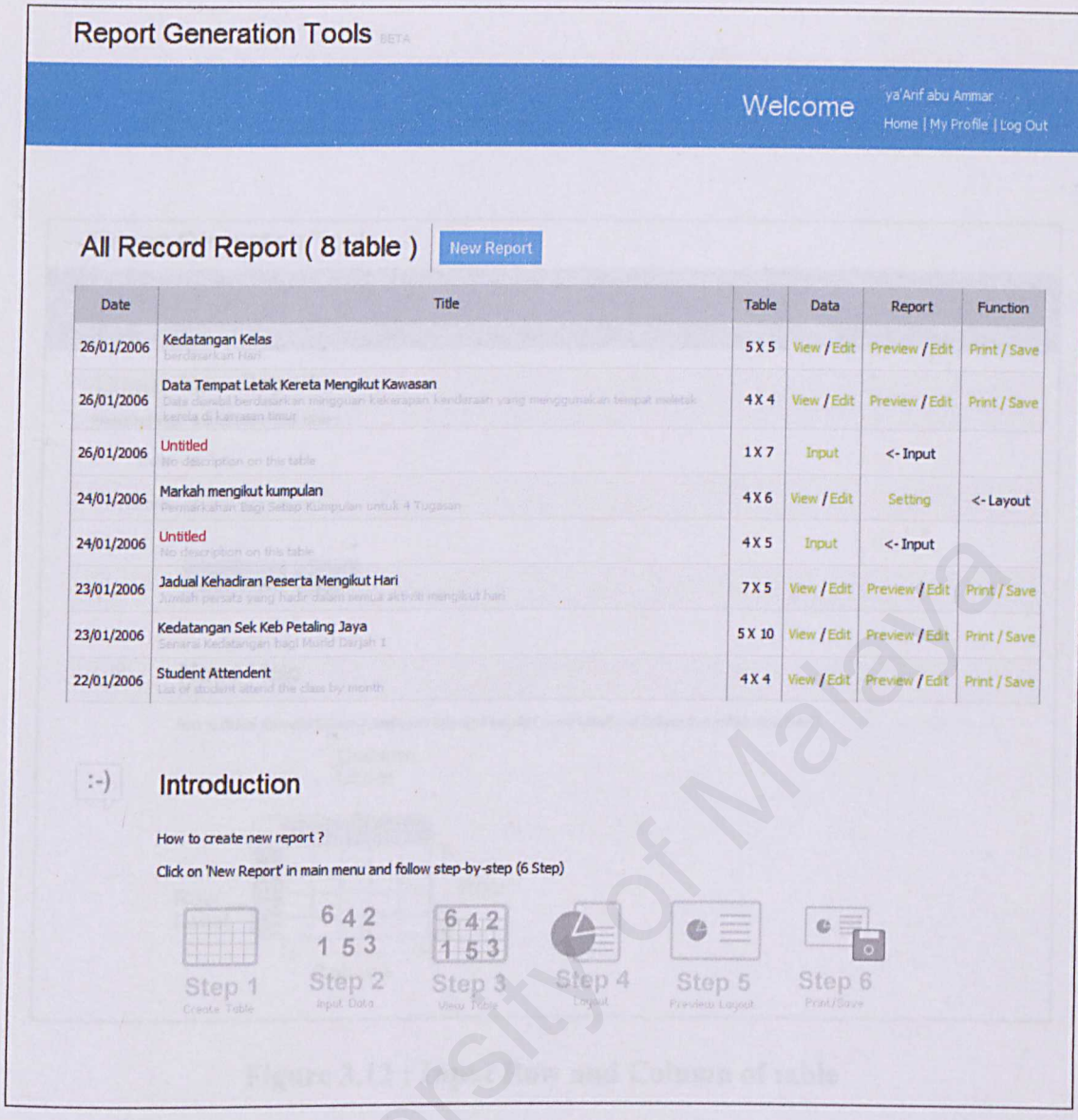


Figure 3.11: Main Page after login success

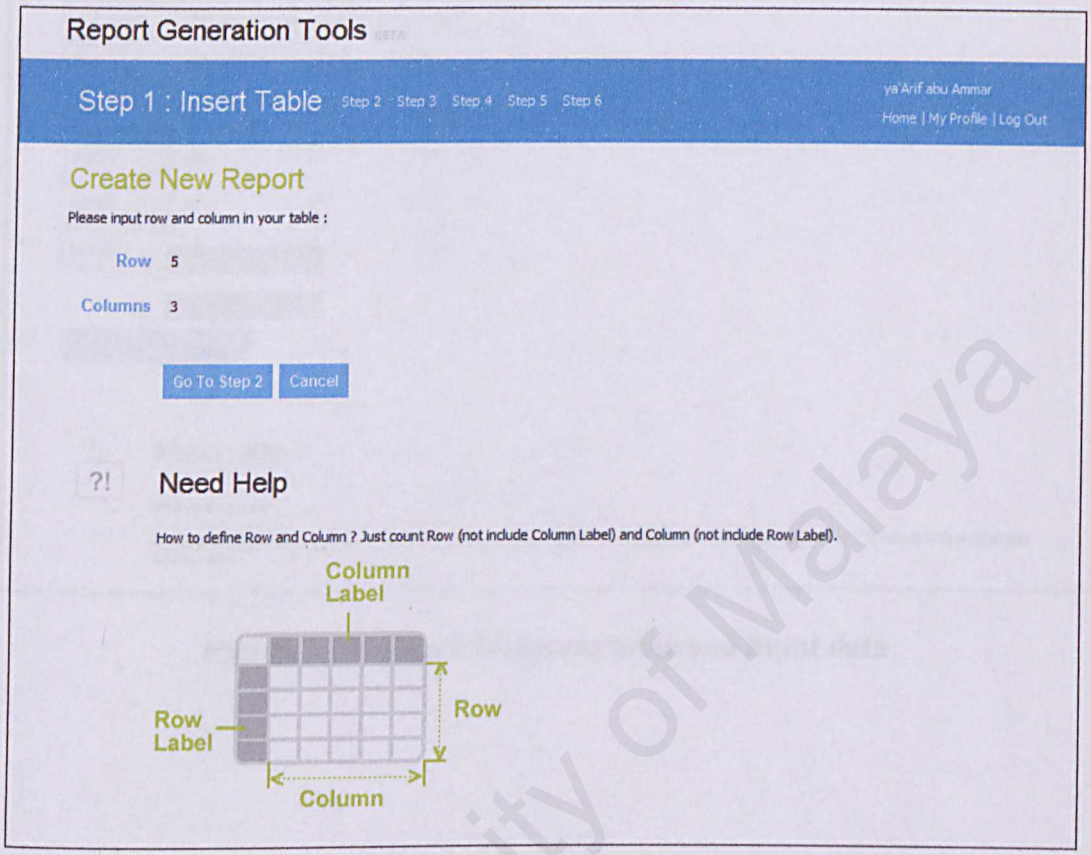


Figure 3.12 : Input Row and Column of table

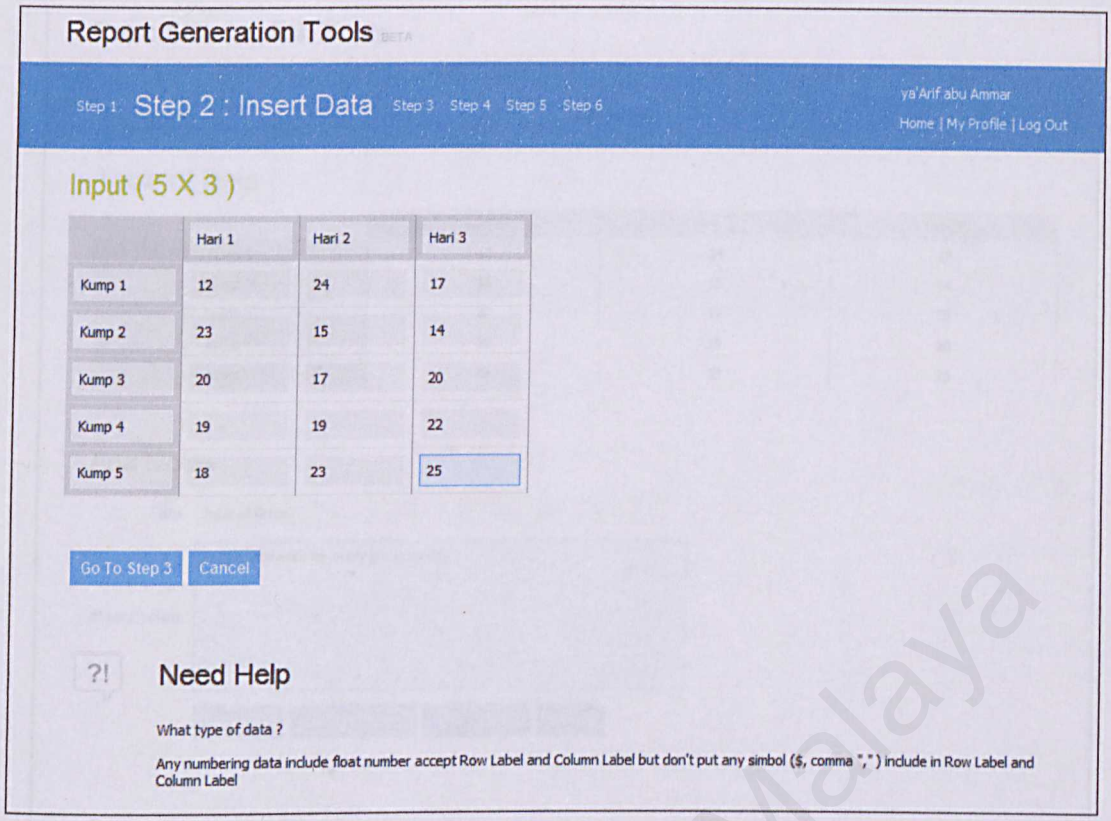


Figure 3.13: Step 2 Generate table and input data

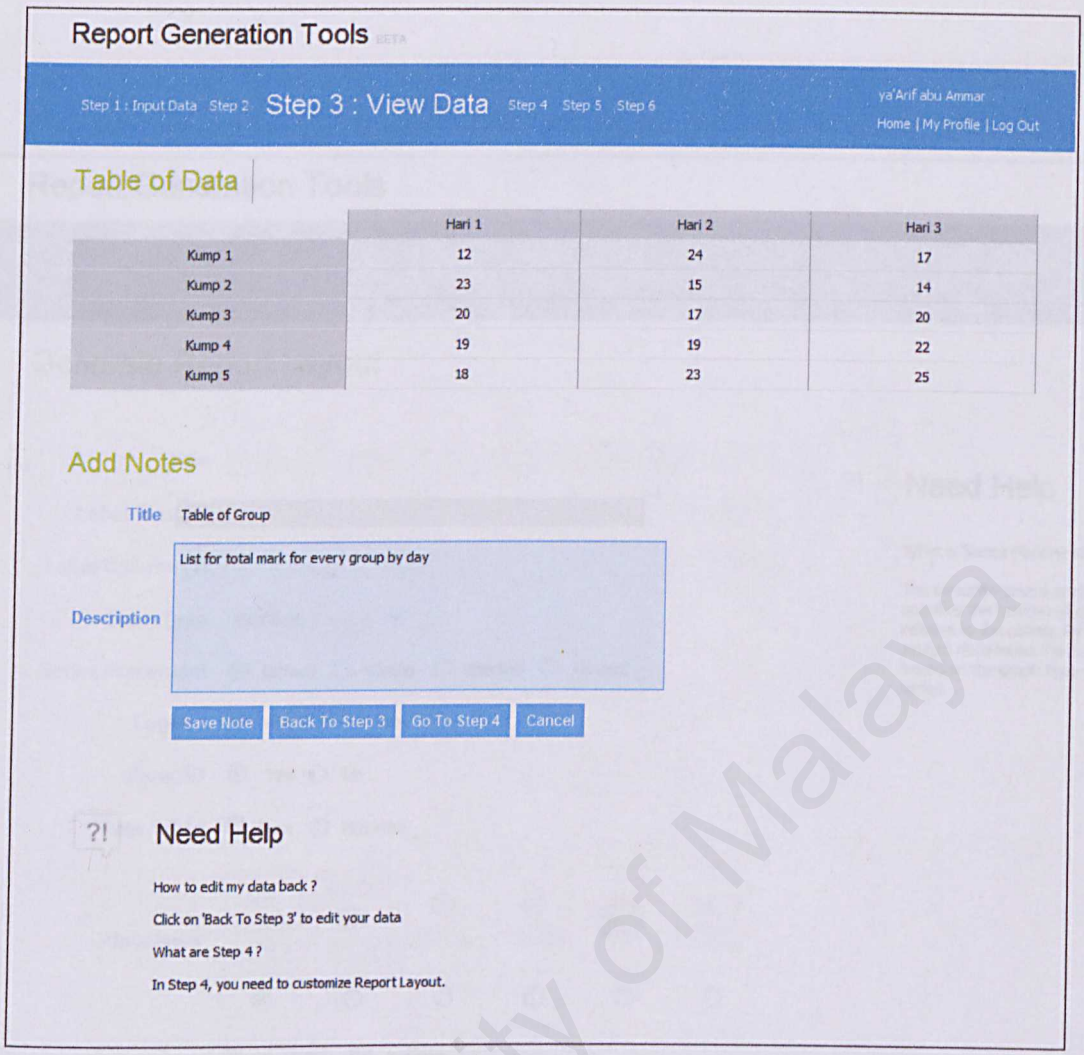


Figure 3.14: Step 3, add title and notes to table

Figure 3.15: Step 4, Setting for layout of report

Report Generation Tools

Step 1 : Input Data | Step 2 | Step 3 | Step 4 | **Step 5 : Preview** | Step 6

ya'Arif abu Ammar
Home | My Profile | Log Out

Preview Report | Edit Layout | Print / Save | Back Home

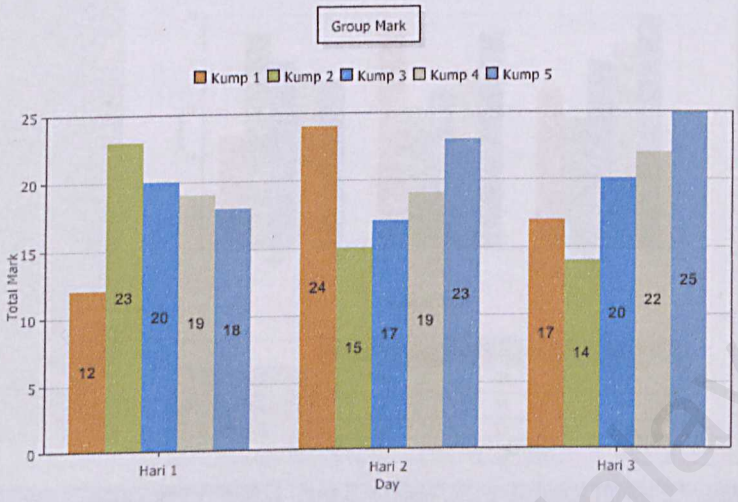


Table of Group

	Hari 1	Hari 2	Hari 3
Kump 1	12	24	17
Kump 2	23	15	14
Kump 3	20	17	20
Kump 4	19	19	22
Kump 5	18	23	25

List for total mark for every group by day

Analysis

Total	288
Maximum Data	25 (1 times at Hari 3, Kump 5)
Minimum Data	12 (1 times at Hari 1, Kump 1)
Average	19.2
Min	19.2
Medium	19
Mod (frequency)	

Figure 3.16: Example report layout with chart, table and analysis

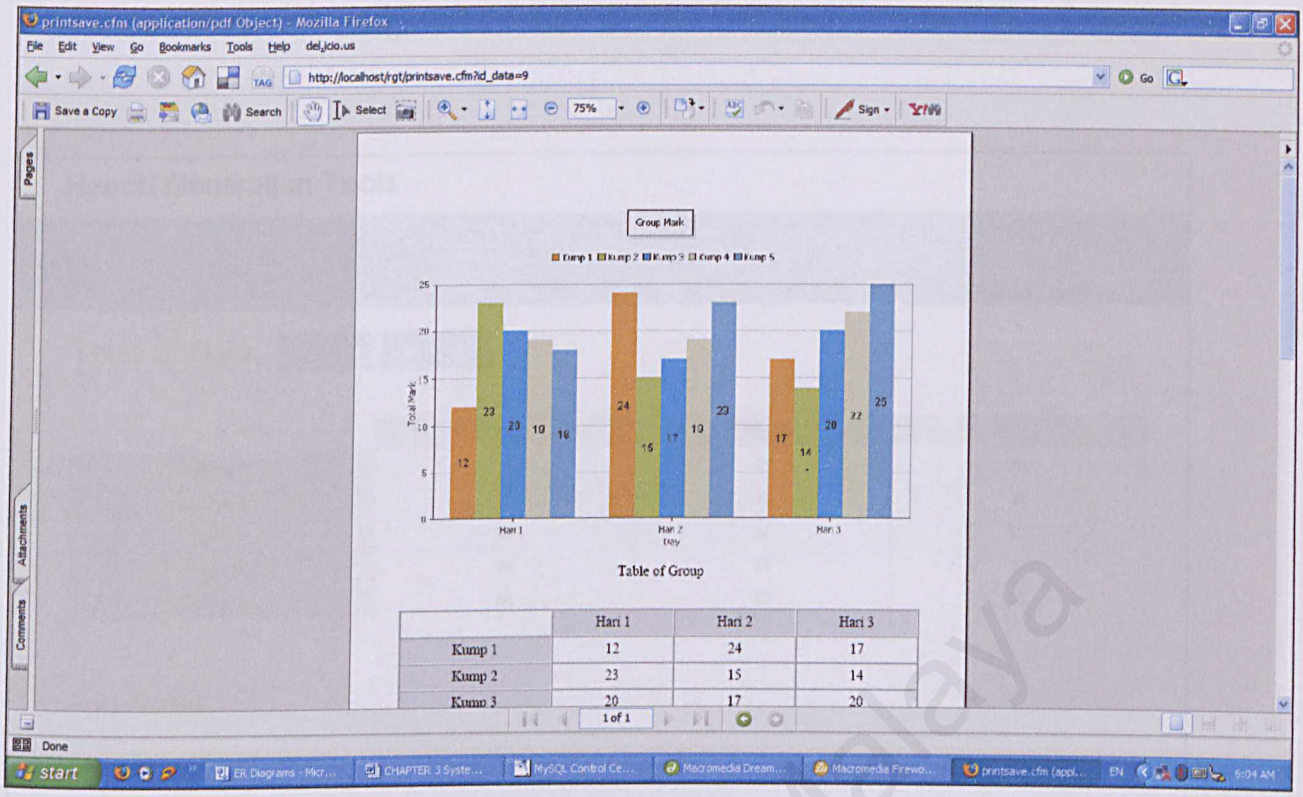


Figure 3.17: Print or Save View in PDF

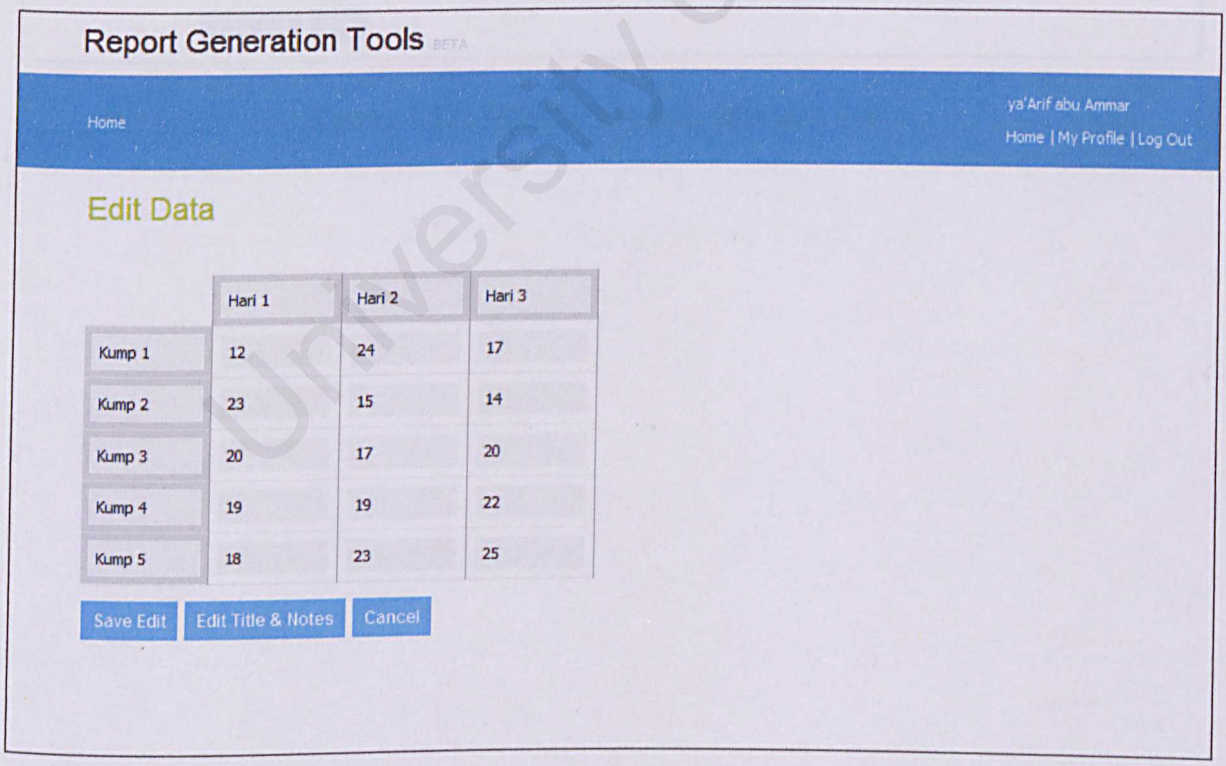


Figure 3.18: Interface for edit data.

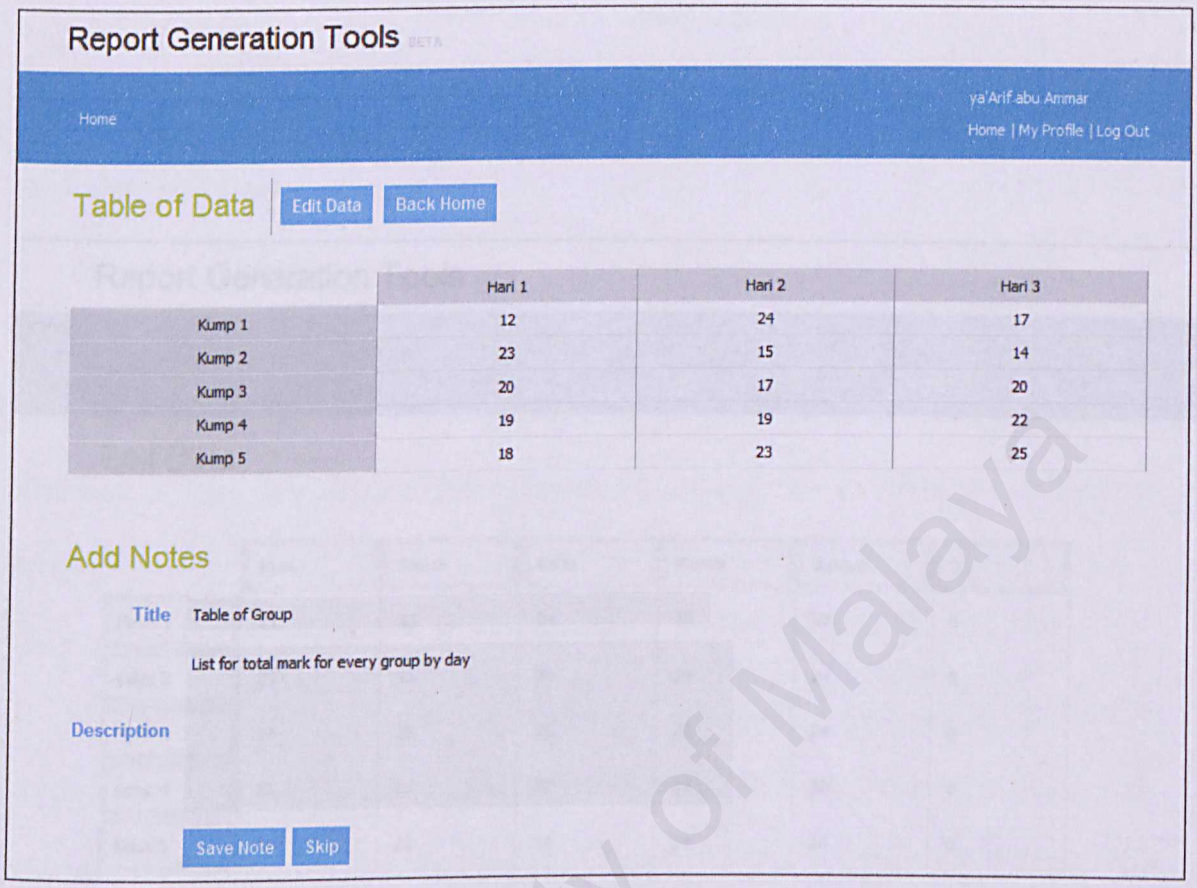


Figure 3.19: Interface to edit title and notes

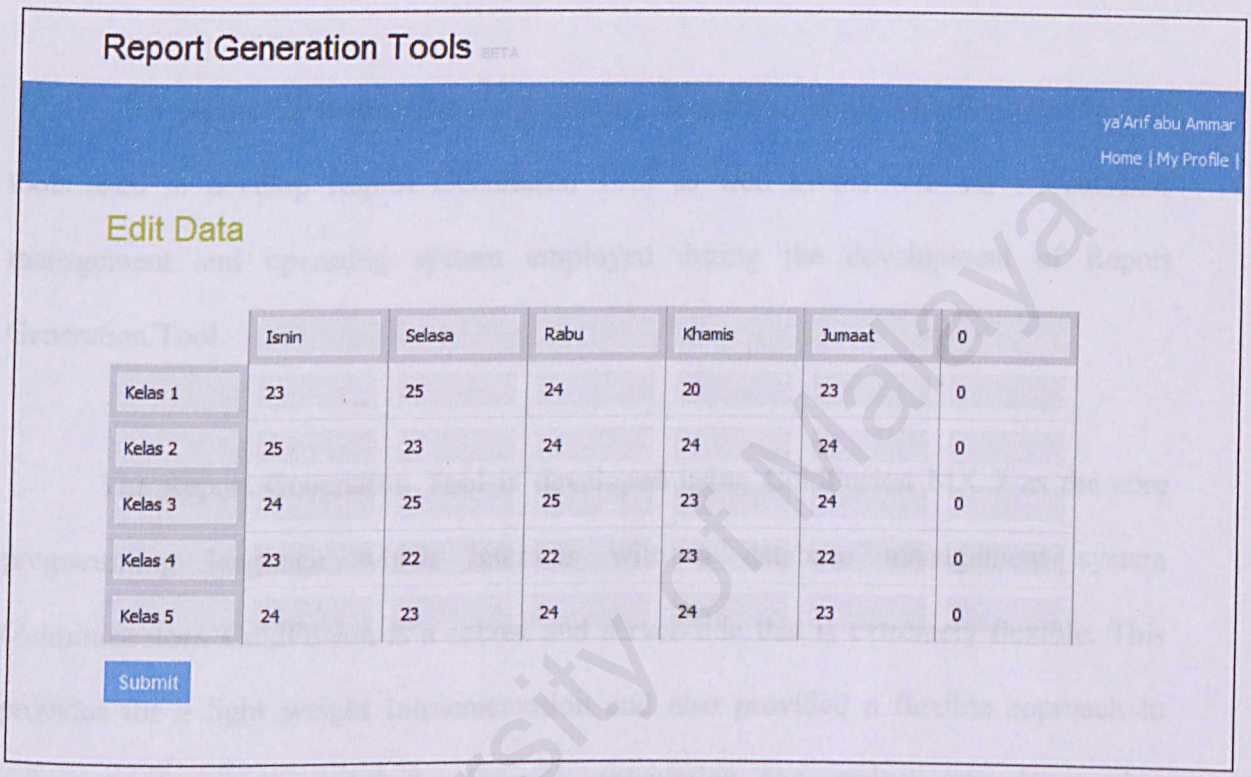


Figure 3.20: add new column or new row in table

CHAPTER 4

System Development

4.1 Development Environment

This section discusses the programming languages along with the development tools used to develop Report Generation Tool as well as the web server, database management and operating system employed during the development of Report Generation Tool.

The Report Generation Tool is developed using ColdFusion MX 7 as the core programming language which interacts with a database management system (Administrator). ColdFusion is a robust and server-side that is extremely flexible. This provides for a light weight implementation and also provided a flexible approach to system implementation. All the formula, calculation and analysis are done using ColdFusion scripting. ColdFusion is also cross platform, which means that the ColdFusion scripts can be executed on UNIX, Linux, or Windows based platform.

The Report Generation Tool consists of code mix of standard HTML, JavaScript and Cascading Style Sheet (CSS). HTML is a markup language used to represent the basic web pages of the Report Generation Tool, while CSS is attached to HTML document, to provide more control over how web pages are displayed and printed. CSS

extend the capability of HTML by providing a richer and more easily updated font, color and layout formatting in HTML documents.

JavaScript is also important which is used to support real-time component calculation. During the development of the Report Generation Tool, JavaScript was used to control permission and alert to user. Moreover, in order to ensure a robust application, the JavaScript language is programmed to implement extensive exception handling. These exceptions are discussed as follows.

1) **Data Fields Validity Checks**

   As part of the system requirement and exception handling ability of Report Generation Tool, the data fields of each form must be validated. Without proper handling, these invalid data fields can completely stop the application of the Report Generation Tool or may cause the system to not be running properly in performing fault calculation and analysis. Moreover, the validity checks functions are employed to ensure the entire data format is consistent and correct when stored in a database.

2) **Invalid User Input**

   It is common for user to perform action or enter data that are not applicable to the application. Therefore, Report Generation Tool must filter out the incorrect input and possibly warn the user of their mistakes. The invalid input format also includes leaving a required field blank in a form.

The database stores the information in an optimized way for quick access by front-end programs. Therefore, the fundamental Report Generation Tool data is held in a standard MySQL database, which is accessed by the ColdFusion Administrator via MySQL connection function library to the database. MySQL is cross platform and robust which makes it reliable to store all the data of Report Generation Tool. MySQL is a thread based memory allocation system, thus it provides fast service with no memory thrashing. In view of the fact that the requirement of Report Generation Tool is to ensure data privacy, MySQL also provides a privilege and password system which is very flexible and secure and allows host based verification. In addition, all password traffic on the net is encrypted.

Web server is a medium to store the Report Generation Tool which process the client's request and logic computing, grab the data required from the database and send the results back to the clients. Thus, it plays an important role because it stores and executes the applications and also processes the requests that have been made. In the development of Report Generation Tool, Internet Information System (IIS) as web server is employed to develop the Report Generation Tool which is running on Windows XP Pro platform.

4.1.1 Development Tools and Technologies Used

1. Macromedia Dreamweaver 8

Dreamweaver 8 is the last version of the program to be produced under the Macromedia banner. It is also the best version to date, providing a streamlined workflow, new tools, more stability, and faster performance. Dreamweaver 8 sports many enhancements to its Cascading Style Sheet support. Dreamweaver 8 is faster and more stable than MX 2004. For example, recreating a site cache, which requires reading every file in your site, is substantially faster in Dreamweaver 8. It also introduces a revamped FTP engine, which is a bit faster, but more importantly, adds a much-requested feature: background FTP. In MX 2004 and earlier versions, whenever Dreamweaver transferred files from your hard drive to your Web server, you had to wait until it was finished before you could do any other work. Dreamweaver 8 introduces a powerful approach to working with XML files: visual XSLT (Extensible Style sheet Language Transformations) authoring (see screenshot at right). XSLT is a language for taking not-so-easy-to-read, tag-based XML files and turning them into a variety of useful formats, including HTML. With this feature, you can convert an XML file into a beautiful Web page, or add formatted XML information to a part of a dynamic page—a PHP, ASP, or ColdFusion page, for example.

Dreamweaver 8 supports efforts to learn and take advantage of new technologies, including PHP 5, Flash Video, ColdFusion MX 7, and the Macromedia Web

Publishing System. Updated support for ColdFusion MX 7 includes new server behaviors and code hinting. To match the code hinting and debugging with the correct version of ColdFusion, Dreamweaver automatically detects the server version the first time it connects to the site. The tight integration between Dreamweaver and ColdFusion lets you add and remove databases directly from the Databases panel, and view only ColdFusion components defined in the current site. For more information, see enabling the ColdFusion enhancements.

2. Macromedia ColdFusion MX 7

Macromedia ColdFusion MX 7 reporting adds integrated business reporting capabilities to ColdFusion, providing unprecedented access to important business data. This new capability enables high-quality, structured reports to be fully integrated into web applications, providing users with well-formatted data that is easy to understand, print, and e-mail. Use integrated business reporting to provide high-quality, structured reports otherwise possible only with expensive third-party reporting products. Users have unprecedented access to important business data in a format that is easily understood. Use the powerful ColdFusion Report Builder to design crisp, well-formatted reports. Include charts and graphs, automatically generate subtotals and totals, and much more. Produce high-quality charts and graphs with over 200 different modifiable attributes to control animation, colors, labels, and more. Use predefined chart styles for professional looking charts and graphs or creates new styles for use across multiple charts and graphs.

Design well-structured, repeating-group report templates quickly and easily with the ColdFusion Report Builder tool included with ColdFusion MX 7. Generate reports within applications in PDF or FlashPaper 2 format, built specifically for viewing documents on the web. Whether creating standard, batch-oriented reports or enabling users to generate reports on the fly, ColdFusion MX 7 reporting offers users unparalleled, structured business reporting power, otherwise only possible using expensive, third-party reporting products.

The ability to print well-formatted content from the browser is poor today at best. Margins are chopped off, page breaks interfere with graphics, and headers and footers are nonexistent for repeating group information. And there is no simple way to save print-formatted web content for offline viewing or to share content by e-mail attachment.

ColdFusion MX 7 changes all that by allowing you to transform existing HTML content dynamically into Adobe PDF or Macromedia FlashPaper 2 formats to produce print-quality output, including headers and footers, multiple page sizes, orientations, ability to save output to disk or e-mail, and more.

For many ColdFusion developers, creating HTML forms is one of the most time-consuming and difficult parts of their job. ColdFusion MX 7 allows developers to

create accessible, high-quality forms in minutes using the new Rich Flash and XML forms capabilities.

Using familiar CFML tags easily create complex, multistep forms with tabbed or accordion interfaces. Applications can use Macromedia Flash controls that are unavailable in regular HTML, such as data grids, tree controls, and calendar date pickers. Best of all, forms will look good and be intuitive for end users without requiring developers to spend extra effort coding presentation-tier information.

3. MySQL

MySQL is consistently recognized for its high quality and reliability. Leading organizations such as Sabre, Yahoo!, Lufthansa, and Cisco depend on MySQL to run their embedded, e-commerce, Web, and data warehousing applications. Plus, independent code quality inspection tests from Coverity, Klocwork, and Reasoning have shown that MySQL's open source approach results in code quality that is up to 6x better than proprietary software. MySQL delivers product reliability and security your company can depend on. Powerful user authentication and authorization, as well as SSH and SSL support ensure you can protect your data assets. MySQL allows you to significantly reduce your database Total Cost of Ownership (TCO). On average, companies using MySQL are experiencing per-project savings of between $250,000 and $500,000. Larger enterprise deployments are yielding savings in the millions of dollars. According

to independent benchmarks, MySQL is a recognized leader in both pure performance and price/performance tests. Query caching delivers significant performance benefits. Flexible replication enables you to cost effectively scale-out your database environment to increase both speed and scalability by easily adding more MySQL servers on low-cost commodity hardware. MySQL enables organizations to cost-effectively scale-out their Web and data warehouse applications using open source software and commodity hardware. Organizations can quickly deliver business-critical applications such as e-commerce, content management, user authentication/authorization, and real-time data warehouses using MySQL. MySQL ease of installation, deployment, and administration is unmatched. You can install and get MySQL up and running in less than 15 minutes, and deliver applications much faster than with other databases. In addition, MySQL requires little or no administration which significantly reduces the time and cost of maintaining your database infrastructure.

4.2 System Implementation

In this development process will show each module that implement in this Report Generation Tool system.

1. **Sign-Up Module**

   Sign-up module provides a registration interface to the user before he or she can make use of the Report Generation Tool as illustrated in **Figure 4.1**. While, **Figure 4.2** shows the exception handling provided by the Report Generation Tool if the username entered by the user during the registration process is already used by another person.

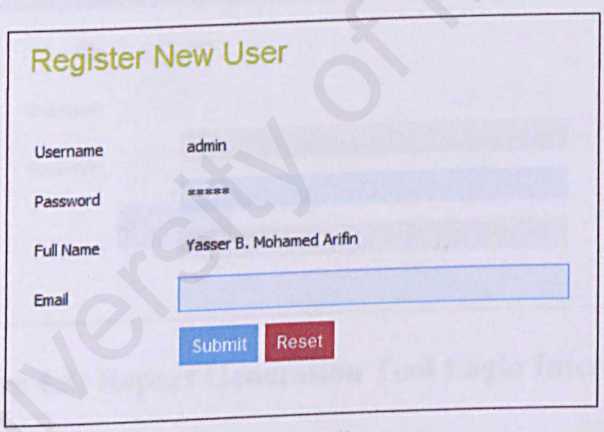


Figure 4.1: Sing-Up Module

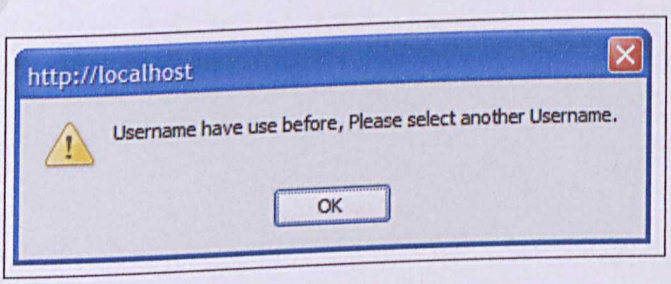


Figure 4.2 : the exception handling for Sign-Up Module

2. Access Authentication Module

The access authentication module addresses the login and logout functions as well as authenticates the accesses and requests made by a computer client as shown in **Figure 4.3**. **Figure 4.4** shows an alert message that will be triggered by the Report generation Tool if a user types in an incorrect username or password. Every user will give one level for determine either this user are administrator or just a usual user.

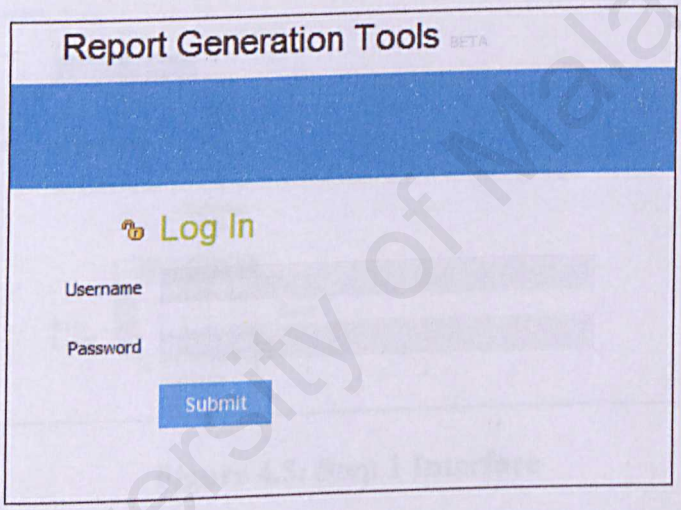


Figure 4.3: Report Generation Tool Login Interface

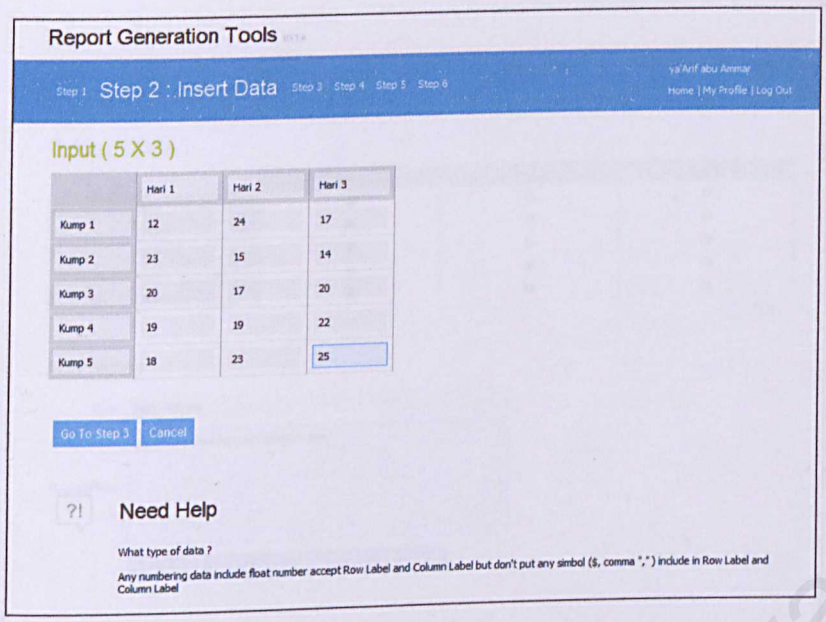


Figure 4.6: Step 2 Interface

Step 3 is optional either to input title or description for table and save it by click at "Save Note" button or just skip to step 4. If user needs to change the input, they can do it by click "Back to step 2" button as show in **Figure 4.7**.

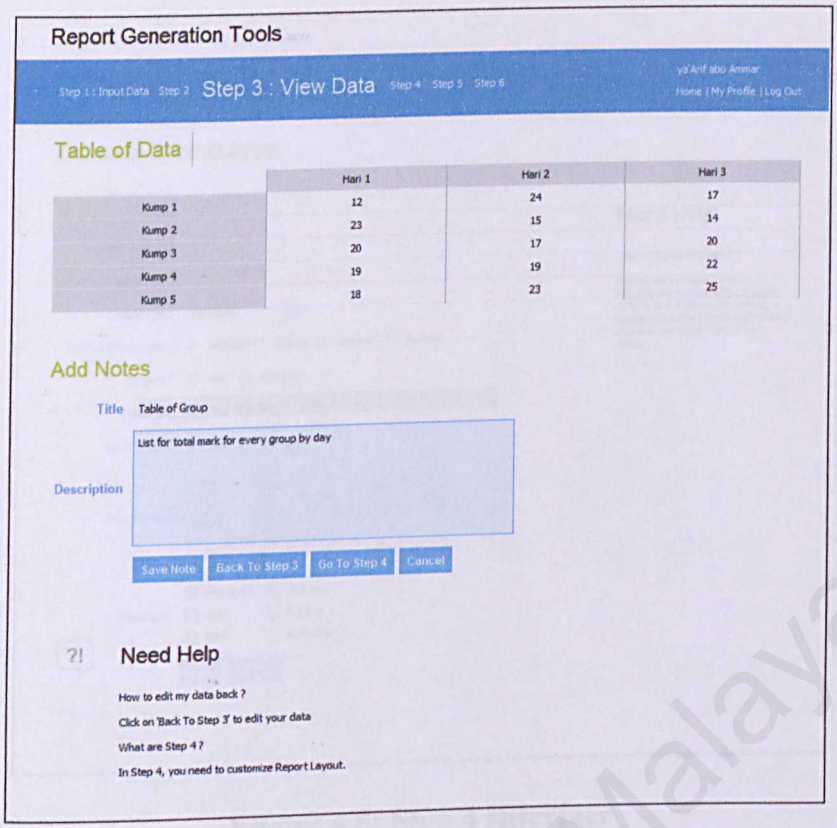


Figure 4.7: Step 3 Interface

Step 4 is layout setting for the final report. In here user need to insert title of chart, row and column label, choose what type of chart, series placement, show the legend or not, show 3D chart or not, view the data table, type of placement, and type of analysis as show in **Figure 4.8**. There are 8 type of chart that can generate from ColdFusion with it will present in flash format (animation) and colorful. For the analysis type, 5 type of analysis that content Max and Min, Average, Min data, Medium data, and Mood data.

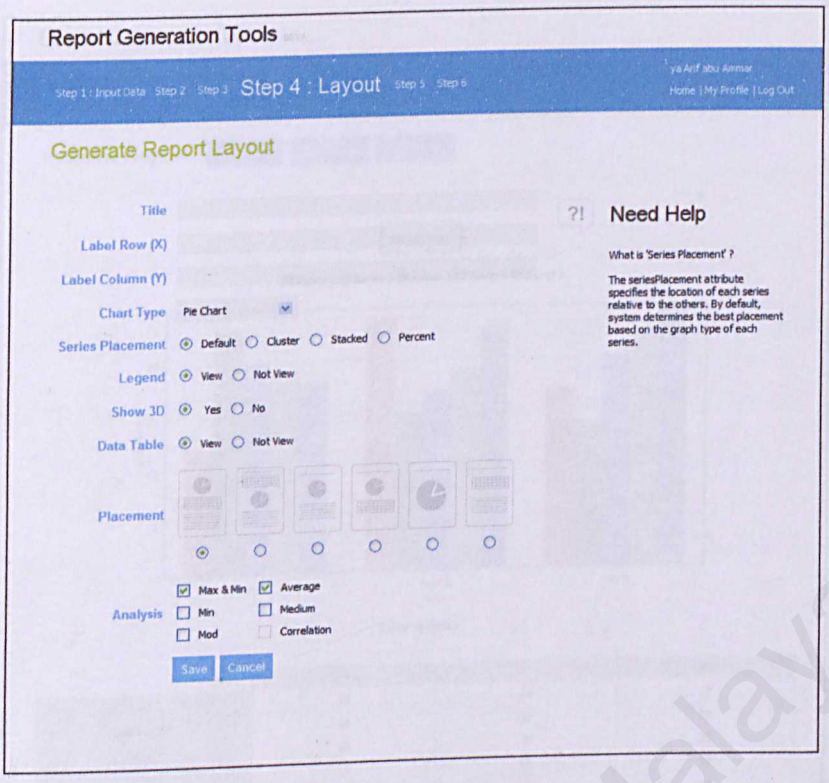


Figure 4.8: Step 4 Interface

Step 5 is a preview of final report that can generate by this Report Generation Tool. All placements of chart, table and analysis are determined in step 4 (see **Figure 4.8**). In this step user also can edit layout by click at "Edit layout" button or go to PDF format by click at "Print/Save" button.

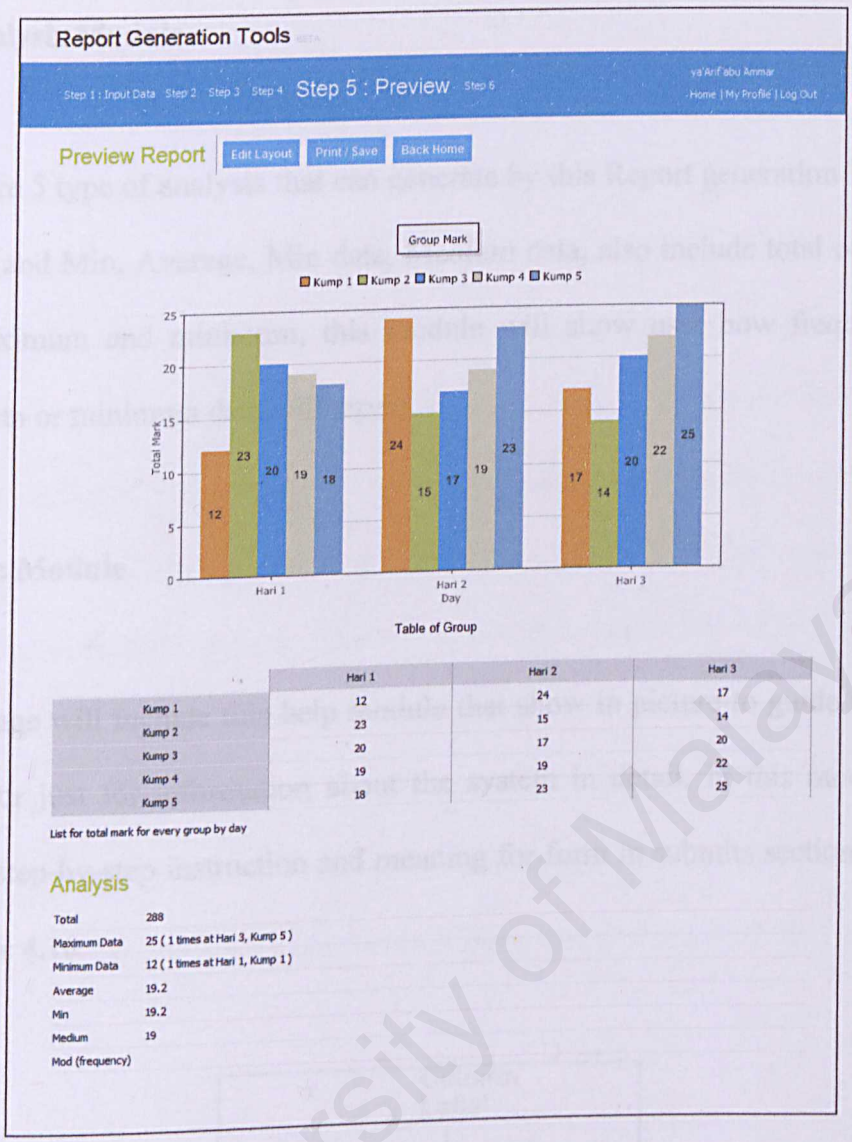


Figure 4.9: Step 5 Interface

Step 6, will bring to final view of report in PDF format that user can save or print the report directly.

4. Analysis Module

There are 5 type of analysis that can generate by this Report generation Tool such as Max and Min, Average, Min data, Medium data, also include total of all data. For maximum and minimum, this module will show user how frequency the maximum or minimum data will repeat.

5. Help Module

Every page will include this help module that show in picture to guide user how to user or just for information about the system in detail. In this module also include step-by-step instruction and meaning for form in submits section as show in **Figure 4.10.**

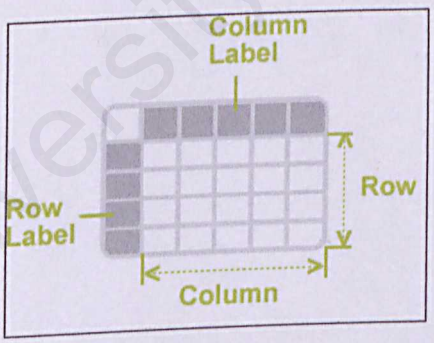


Figure 4.10: Example Help Module

4.2.1 Design the Programs

This phase will combine all images and graphics to be one complete system. This phase take time longer than other phase because the processes a bit complicated and need to be done conscientiously. System being builds with writing the program codes based on design of the system. Overall of this package does not need being programmed because Macromedia Dreamweaver 8 is software that gives many advantages to built graphic interface without needing any script or code. It has been automatic generate with images, graphics or animations that been put on the stage with no coding at all. A few aspects have to take concern to write code:

- Write code that easy to read, easy to replace and not too complex.
- Coding that being used must be standardizing. As example; variables that being used must show certain function and being defined at the early program.

4.2.2 Coding Styles

There two type of coding style that use in this Report Generation Tool. First, separate coding with the user interface that mean interface design and function are in the deference file. As example, this system will call table or chart function in deference place for generate table or chart in report.

Second, coding and user interface in the same file that mean in the top of syntax are function that need to do when call this file and the bottom are syntax to generate user interface. As example, syntax to insert the data to database that include in same file name when user call/open this interface.

4.2.3 Testing and Debugging

Macromedia ColdFusion MX 7 are the best compile to testing and debugging the syntax only by using web browse or just testing and debugging in Macromedia Dreamweaver 8. In Macromedia ColdFusion testing and debugging will show number of line syntax error are located and what should it be and the example of syntax function for ColdFusion.

CHAPTER 5

System Testing

5.1 Testing Techniques

Testing is a critical phase of software quality assurance in a software development life cycle. The objective is to execute a program with the intention to discover errors, evaluate the systems' abilities and most importantly, to determine whether it complies according to its requirements and functional specifications.

During the development of Report Generation Tool, after completing the coding phase, testing began from individual program modules and progressed towards the entire system. There are five stages in the testing phase of Report Generation Tool; unit testing, module testing, integration testing, system testing and security testing.

Test environment is important to ensure that the testing range is from the minimum to the recommended for the client hardware and software acknowledged in the user requirements. Table 5.1 illustrates a list of browser specification and platform which the Report Generation Tool application tested on.

Table 5.1: Test Environment of Report Generation Tool

Platform	Web Browser
Windows Based	1. Internet Explorer 5.X and higher 2. Netscape 7.X and higher 3. Opera 7.23 4. Mozilla Firefox 0.9.X
MacOS Based	1. Internet Explorer 5.2.3 2. Netscape 7.1
UNIX Based	1. Netscape 7.X and higher

The screen resolution during testing was 800 x 600 pixels and 1024 x 768 pixels with color depth of 32-bits. Report Generation Tool was not tested other than web browser, platform, screen resolution and color depth stated above. The testing used personal computer with machine specification of Pentium 4 with 512 Megabyte of RAM and Pentium M Centrino with 512 Megabyte of RAM.

5.1.1 Unit Testing

Unit testing is conducted to ensure that proper functionality and code coverage of Report Generation Tool have been achieved during coding. This stage of testing verifies that each component functions correctly with proper input and output expected based on the component designs and requirements. During the development of Report Generation Tool, there were several steps being carried out which are:

- Code review which intended to examine the correctness of program codes by reading line by line and attempt to spot algorithm, data and syntax faults.
- Performs a test to each button and link to ensure that it functions as required
- Develop test cases to show the input is properly processed to expected output.
- Boundary conditions are tested to make sure the functions run at boundaries established for limiting process.
- Test all errors handling paths.
- Database including tables, indexes and connection.

The exit criterion for this milestone is code-complete. All functionality and logical and physical components of the application were completed and made available for module testing.

5.1.2 Module Testing

A module is a collection of dependent components unit which encapsulates related components unit. Testing can be carried out on every module that was defined in the requirement phase. Each module is tested to ensure that the desired functions can run successfully and as expected, and also to verify the correctness of the flows of events.

5.1.3 Integration Test

The integration testing of the Report Generation Tool can be viewed from an incremental perspective, whereby the test starts from the main page of Report Generation Tool and moves down to the lowest level of the functions. Using this approach the software is built piece-by-piece and tested little-by-little. For the Report Generation Tool, this test is also used to ensure that all the modules are connected and linked according to the requirements in order to discover errors associated with the interface.

5.1.4 System Testing

System testing in Report Generation Tool involves testing against integrated hardware and software system in order to verify that the system meets the specified requirements as described in the requirement specifications. Also, it involves a series of different test designed to fully exercise the system to uncover its limitation and measure its capabilities. The Report Generation Tool system testing takes place at a higher level, whereby the testing focuses on behavior rather than functional structure.

This system has to be tested to:

- Make sure all sub modules can interact one another without any achieve conflict at any sub module.
- Cover integration and cooperation among the software and hardware being built.
- Testing if recovering process can be done quickly if there errors.
- Testing if safety control can be trusted and fulfill.
- Testing if system development consistent with what has been specified.

5.2 Testing plan

Testing plan is for design and organizes structured testing activities. It can make sure testing processes done perfectly and regularly. Also cover all thing that need to done. Steps in testing plan:

- Recognize testing objectives
- Design testing case
- Write testing case
- Test testing case
- Do the testing
- Evaluate testing result

5.3 Maintenance

Any changes happen at function and sub modules need maintenance at all development phase maintenance process done to make sure all system function can support any changes happen to sub modules or functions in this package. Maintenance processes focusing on system performance divides into four main aspect:

- Maintenance control over system function
- Control of the system changes
- To make sure function can be accept and truly complete
- Maintain the system performance consistently.

CHAPTER 6

Discussions and Conclusion

6.1 Problems Encountered and Solutions

In develop this Report Generation Tool; developer had faced a few problems in system development phase. Developer had attempted to solve the problems, so the system can be developed within the time.

1.1.1 Use of New Development Tools

For new in ColdFusion technology, there are many things that need to learn. First, how to configure Macromedia ColdFusion MX 7 with Macromedia Dreamweaver, second how to make connection between Macromedia ColdFusion MX 7 with MySQL database, third to make configuration between Macromedia ColdFusion MX 7 and Internet Information System (IIS).

Solution for the all this problem is to try to setting configuration between these applications in ColdFusion. Developer just needs to go to Administrator as show in **Figure 6.0** and make little change between database and localhost.

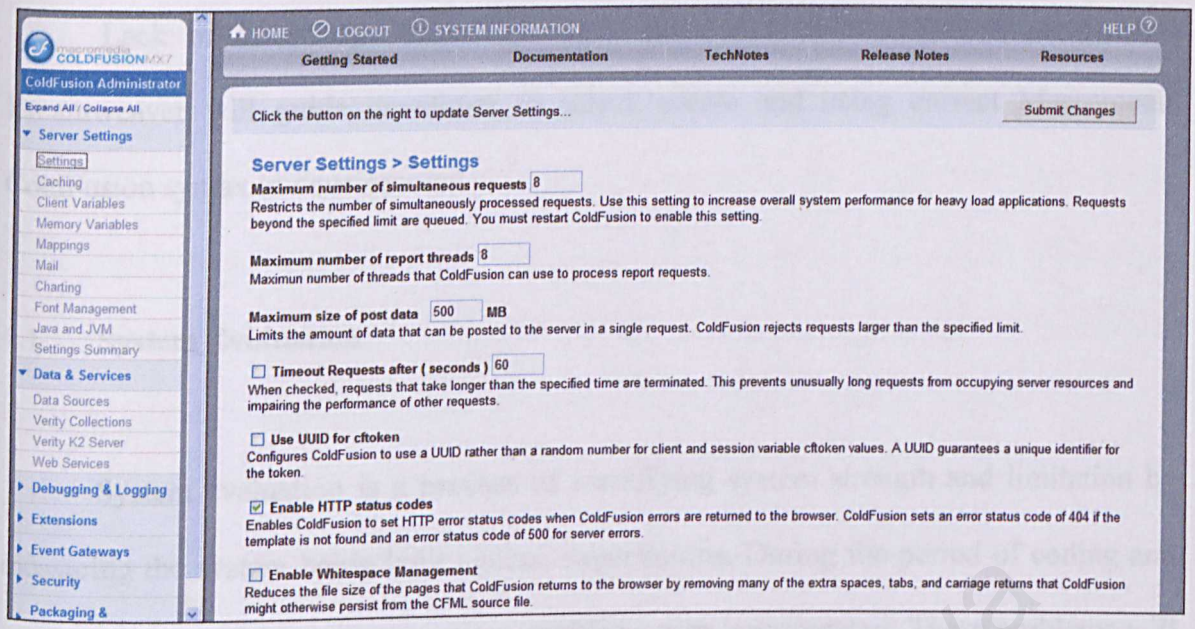


Figure 6.0: ColdFusion Administrator

1.1.2 Lack of Programming Skills

When using Macromedia ColdFusion MX 7, developer need to explorer more detail about this new programming technology. How to use function that has implemented in this ColdFusion and how to make relation between Macromedia ColdFusion with another programming language like JavaScript, ActionSript, and etc.

Macromedia ColdFusion gives an advantage to new developer to easy and fast to create dynamic application by just using function that has already include in ColdFusion such as CFCHART to create a complete chart by using Macromedia Flash format or JPG format with colorful and animation.

Lack of knowledge about syntax is not big problem because Macromedia Dreamweaver will guide developer to select, create and using correct Macromedia ColdFusion syntax in development.

1.1 System Evaluation

System evaluation is a process of identifying system strength and limitation by measuring the system being built against expectations. During the period of coding and implementation of this system, various problems were encountered. These problems will solve, through references on notes documentation, notes user discussion groups and discussion among friends. These steps have helped me to identify the system's strength, limitations and enhancements.

6.2.1 System Strengths

Evaluation of the system strength is focused on the strength of Macromedia Dreamweaver 8, Macromedia ColdFusion MX 7 and other used software. By developing with all these tools, this system inherited the strength of being able to execute in multiplatform as well as being able to deliver a dynamic and interactive content.

For Report Generation Tool, evaluation of the system strength is this system can be accessed through any operating system platform such as Windows NT, Windows 98 and later, Apple Macintosh, Linux and etc. Beside that user don't need to have any plug-in application to run this program but just need web browser that supports new technology in flash and etc.

The Report Generation Tool is developed in about three months. Speed rapid development is important because system that requires a gestation to develop may be out dated even before it ready to use.

The Rapid Generation Tool that was developed is simple and user-friendly. Users who are familiar with Graphics User Interface (GUI) should have no problem at all in browsing and using the system.

1.1.1 System Weaknesses and Limitations

This Report Generation Tool is not so complex compares then Crystal Report tool because Report Generation Tool can not call any database from client or any source. This system need to user input all data back into the database so it will take more time and more work to get report.

This system only can accept numbering data type that means this system can not analysis string data or etc but it can separate between label and data in array function. For example, if user input Monday, Tuesday, Wednesday, Thursday, Friday, Class 1, 23, 24, 22, 23, 23 Class 2, 22, 23, 23, 24, 23 so this system will arrange the data in to table as show in **Table 6.1** below.

Table 6.1: Example data table.

	Monday	Tuesday	Wednesday	Thursday	Friday
Class 1	23	24	22	23	23
Class 2	22	23	23	24	23

Other limitation for this Report Generation Tool is can not compare more then one table that user has input. The report that generate only can analysis data from one table but can not compare between two or more table to get statistic or etc. Only some placement of output that user can select to generate report. So the output will look like static.

6.2.3 Future Enhancements and Expansions

For the future enhancements in Report Generation Tool, this system should have to connect or call any database like Microsoft Office Access, MySQL, Oracle or etc. this system also need to make comparison between two or more table to get more specific analysis.

Analysis need more variety like make some conclusion or decide the best result or can count some bigger data.

6.3 Conclusion

Report Generation Tool are simple report generate that include verity of chart, simple analysis and can convert into PDF file. This system sill is a good system if it can make connection with another database. With the 6 step to generate one report can make user easier to learn and more faster then other report generation. If user forget or skip one of these steps, this Report Generation Tool will tell the user to back and make a changer to get a good report. This system is normal user-friendly system, it has more specific purpose. It help user to generate report faster and easy with online application.

REFERENCES & BIBLIOGRAPHY

1. Shneiderman B., Plainsant C., U.S.A, *Design The User Interface*, 4th Edition. 2005
2. Peter F. Marina S., Russia, *System for Database Reports Generating*, Page 84 - 88 vol.1, 2001.
3. Daniel K.C., France, *a Document-driven Approach to Database Report Generation*, Page 925 – 930, 1998.
4. Paul G., Jeffery S., *Automated Rover Sequence Report Generation, California*, Page(s):7 - 3337 vol.7, 2001.
5. Oracle Corporation, U.S.A. *Building Reports Manual*, 2.5 editions, Apr. 1995.
6. Nigon T., *A Method of Reporting Distribution Field Inspection Data*, Page(s) A3 - 1-6, 1997.
7. http://www.macromedia.com
8. http://www.mysql.com

Appendix A

Questionaire

Fakulti Sains Komputer & Teknologi Maklumat
University Malaya

Pengenalan

Tinjauan ini dilakukan bagi mendapat maklum balas pengguna/masyarakat mengenai penggunaan sistem secara 'online' dan ia juga bagi memudahkan dalam proses menentukan keperluan sistem.

Bahagian 1 : Personal Information (* tandakan ✓ pada yang berkenaan sahaja)

Umur ☐☐ Tahun

Jantina * ☐ Lelaki ☐ Perempuan

Pekerjaan ☐

Tahap Pengajian * ☐ STPM ☐ Diploma ☐ Ijazah ☐ Lain-Lain

Kekerapan Penggunaan Internet * ☐ Setiap hari ☐ Mengikut keperluan ☐ Beberapa kali dalam seminggu ☐ Tidak pernah

Bahagian 2 : Sistem yang sedia ada

Pernah menggunakan Ejen pencarian * ☐ Ya ☐ Tidak

Ejen pencarian yang digunakan * ☐ Google ☐ Yahoo ! Search ☐ Lain-lain, nyatakan ☐

Pencarian berdasarkan * ☐ Perkataan ☐ Ayat ☐ Lain-lain, nyatakan ☐

Penghasilan laporan menggunakan * ☐ Perisian ☐ Secara Talian (online)

Jenis laporan * ☐ Formal ☐ Tidak Formal

Pernah menggunakan Ejen pencarian * ☐ Ya ☐ Tidak

Ejen pencarian yang digunakan * ☐ Google ☐ Yahoo ! Search
☐ Lain-lain, nyatakan ☐

Pencarian berdasarkan * ☐ Perkataan ☐ Ayat
☐ Lain-lain, nyatakan ☐

Penghasilan laporan menggunakan * ☐ Perisian ☐ Secara Talian (online)

Jenis laporan * ☐ Formal ☐ Tidak Formal

Maklumat dalam laporan * ☐ Peratusan ☐ Maklumat ringkas
☐ Jadual ☐ Graf atau Carta Pai
☐ Data ☐ Maklumat laporan
☐ Lain-lain, nyatakan ☐

Bahagian 3 : Keperluan Ejen Pencarian dan Penghasilan Laporan

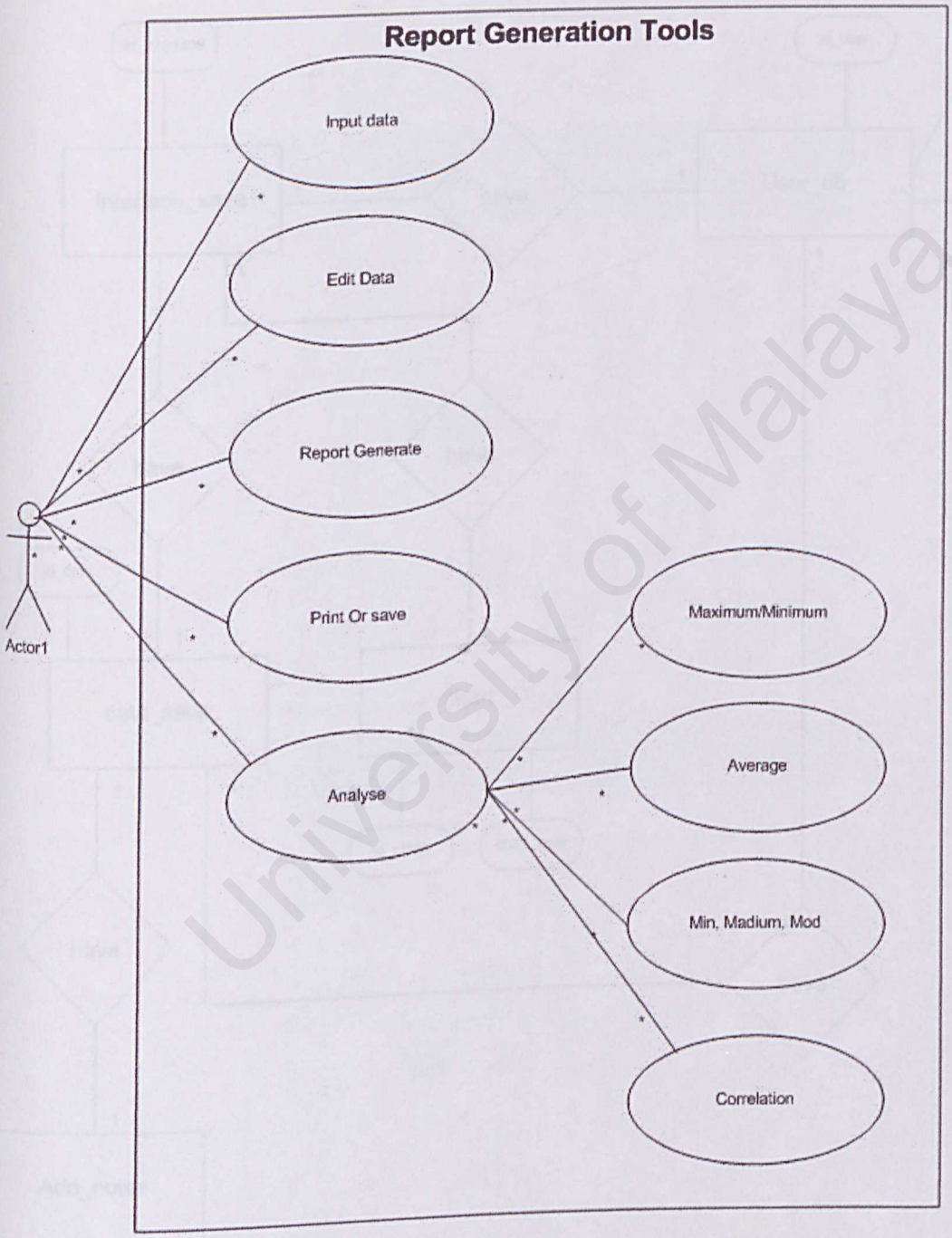
Jenis carta laporan * ☐ Carta Pai ☐ Graf Garis / Bar
☐ Jadual ☐ 'Stock'
☐ 'Area' ☐ Titik XY
☐ Lain-lain, nyatakan ☐

Pencarian berdasarkan * ☐ Perkataan ☐ Ayat
☐ Istilah ☐ Kata Kunci
☐ Lain-lain, nyatakan ☐

Cadangan ☐

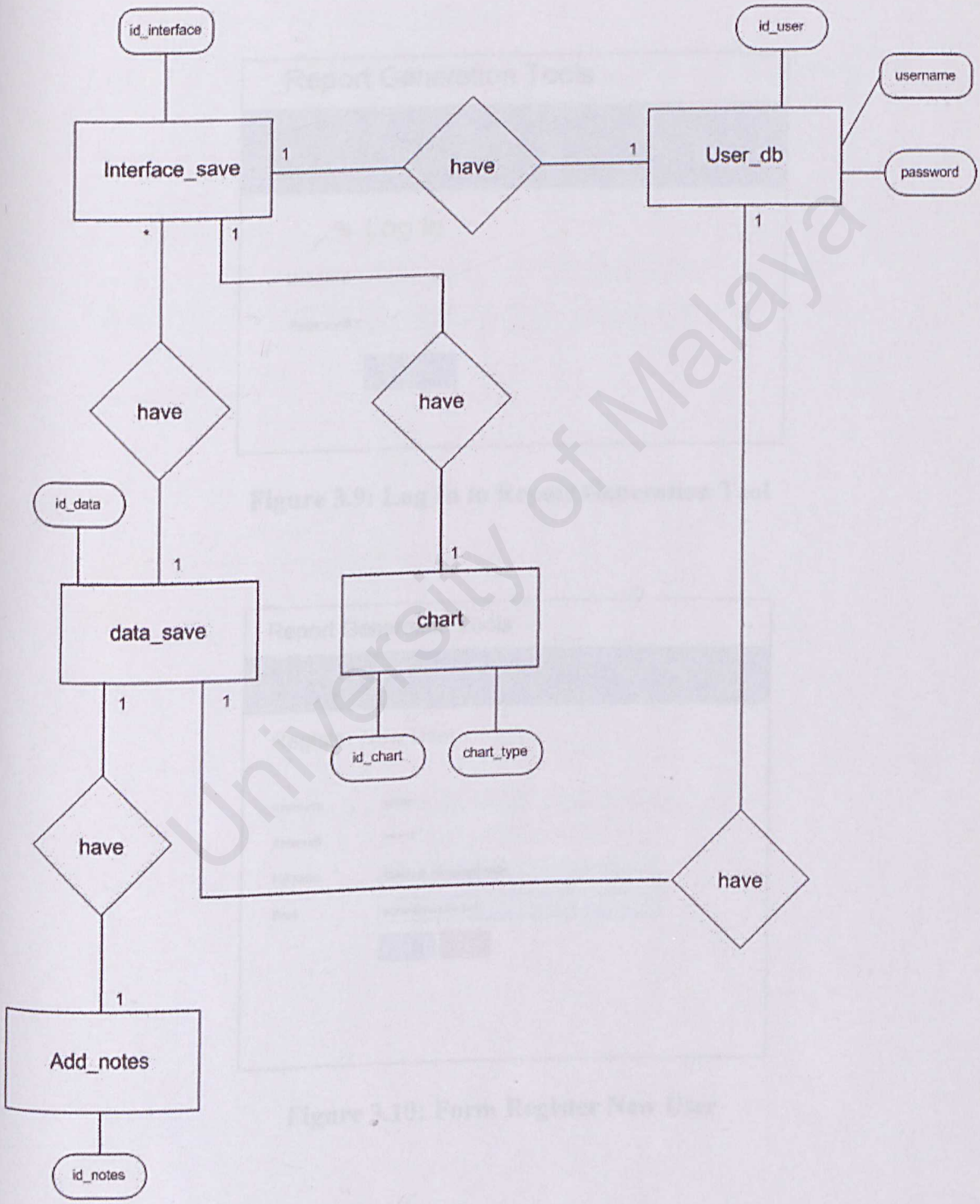
Appendix B

Use Case Diagrams



Appendix C

Database Design



Appendix D

Screen Designs

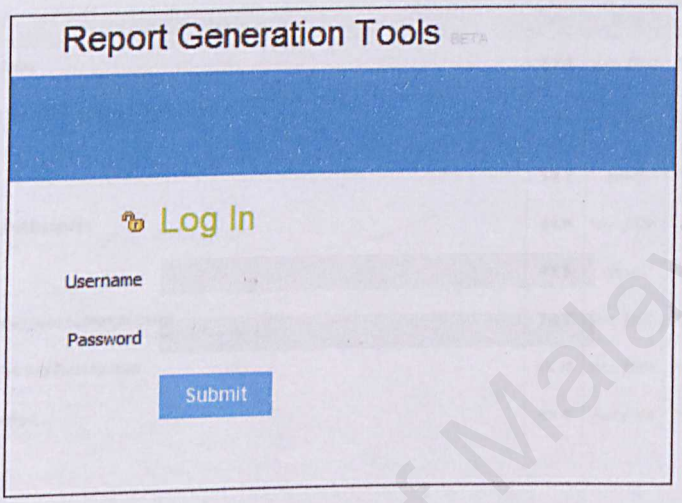


Figure 3.9: Log In to Report Generation Tool

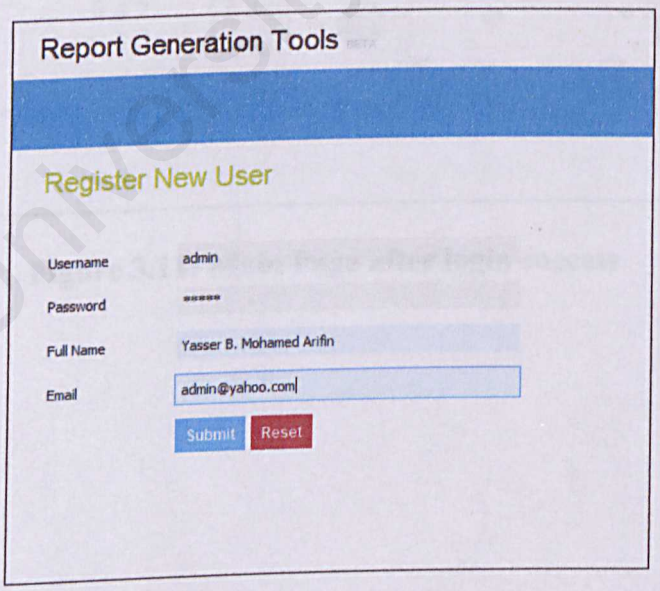


Figure 3.10: Form Register New User

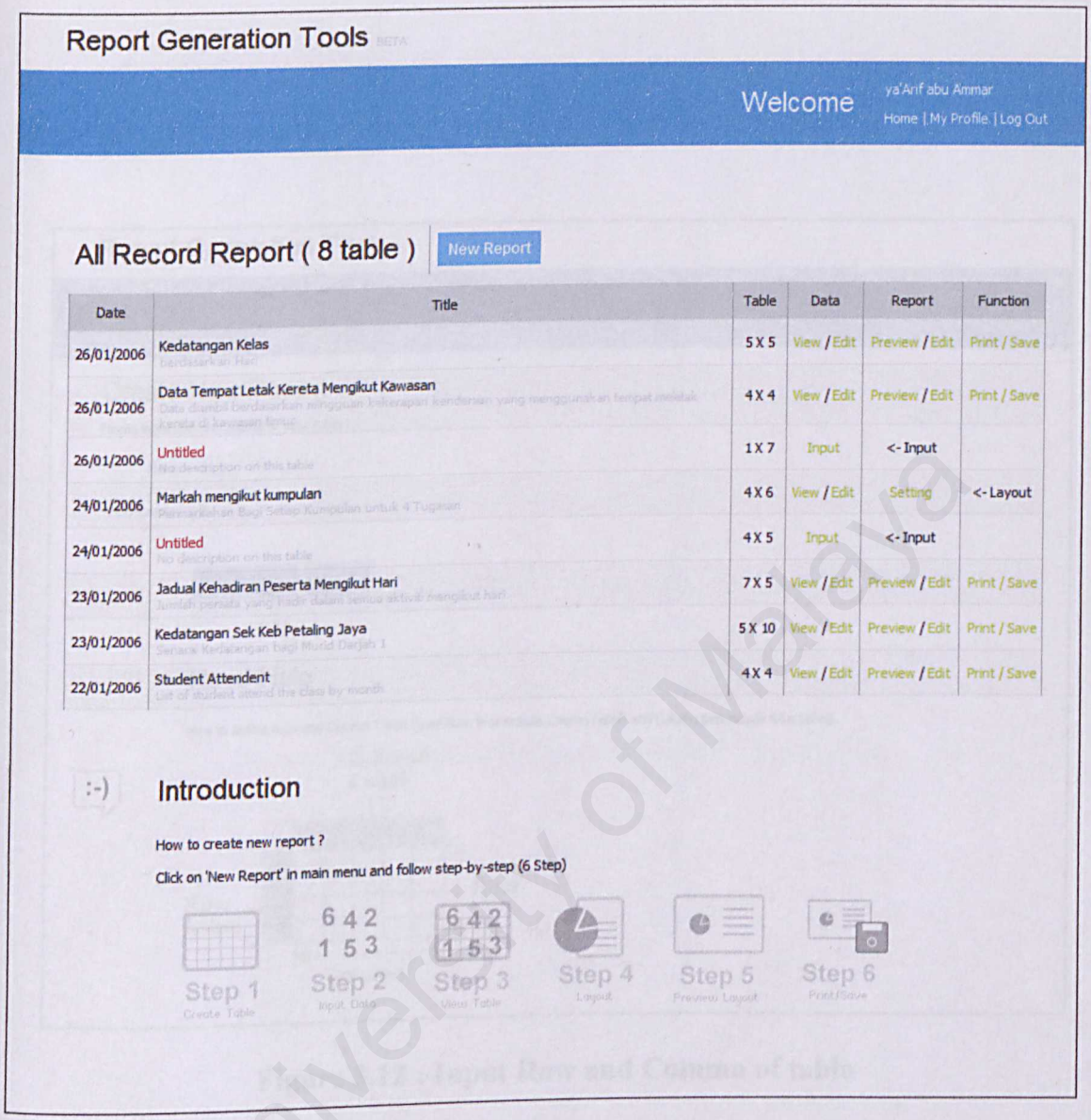


Figure 3.11: Main Page after login success

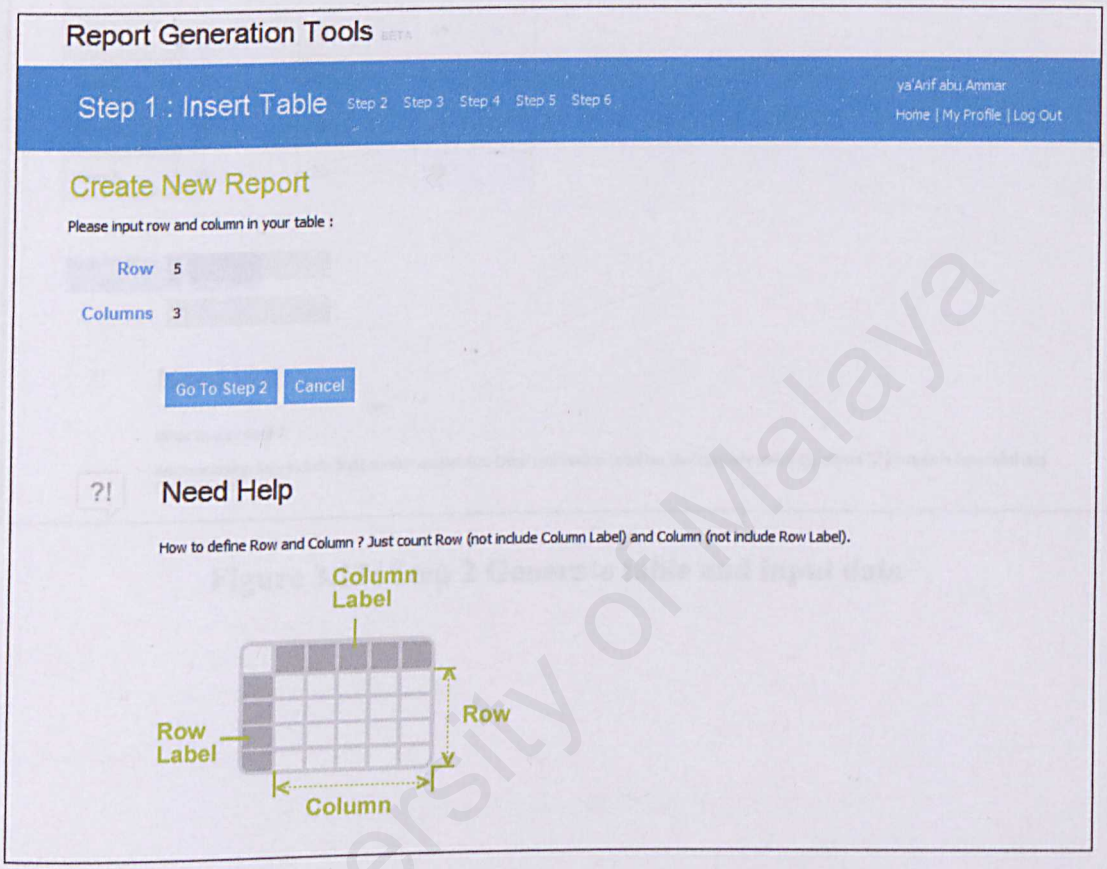


Figure 3.12 : Input Row and Column of table

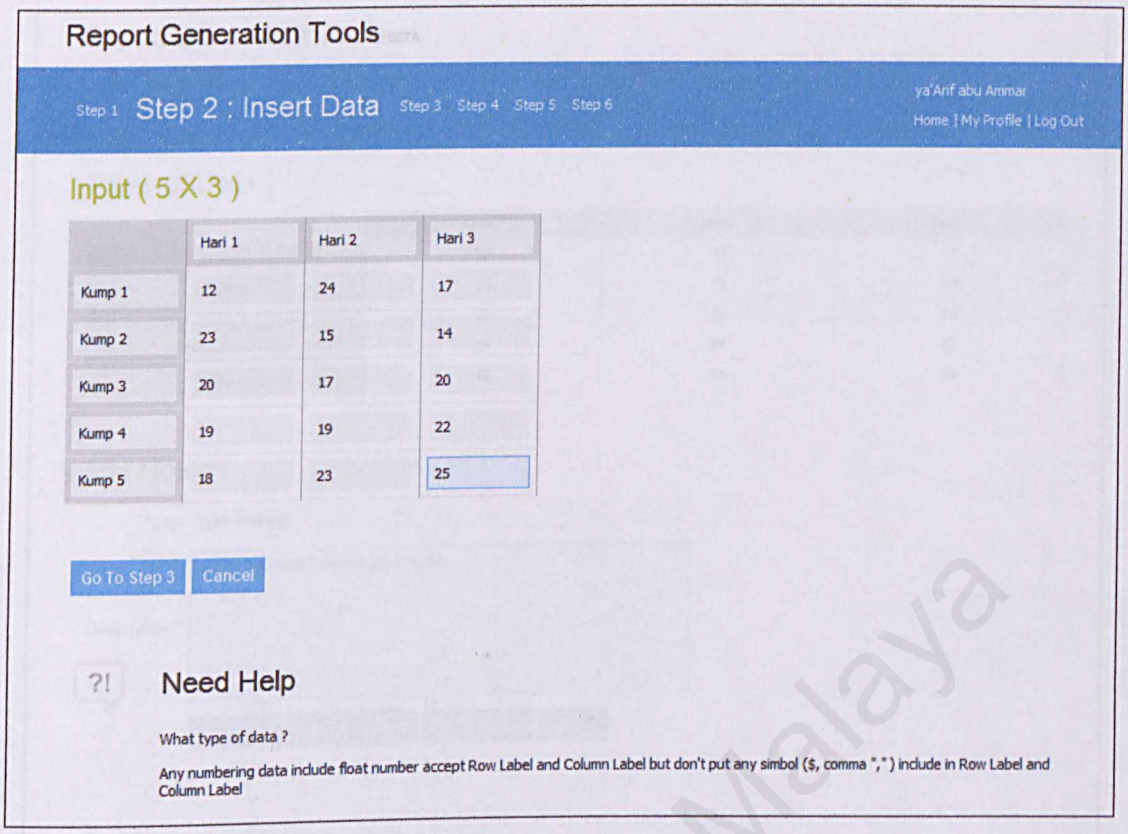


Figure 3.13: Step 2 Generate table and input data

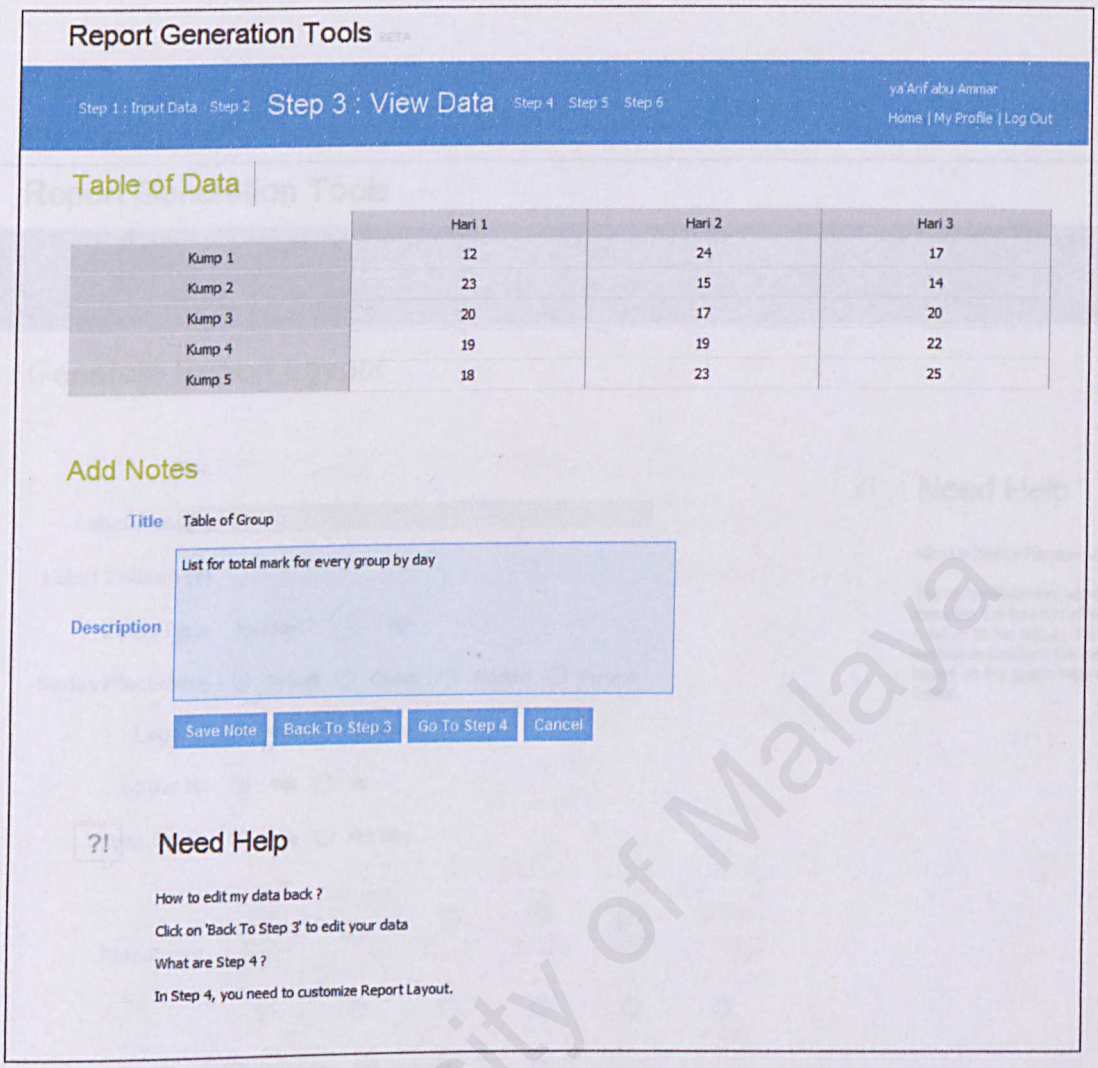


Figure 3.14: Step 3, add title and notes to table

Figure 3.15: Step 4, Setting for layout of report

Report Generation Tools

Step 1 : Input Data Step 2 Step 3 Step 4 **Step 5 : Preview** Step 6

ya'Arif abu Ammar
Home | My Profile | Log Out

Preview Report | Edit Layout | Print / Save | Back Home

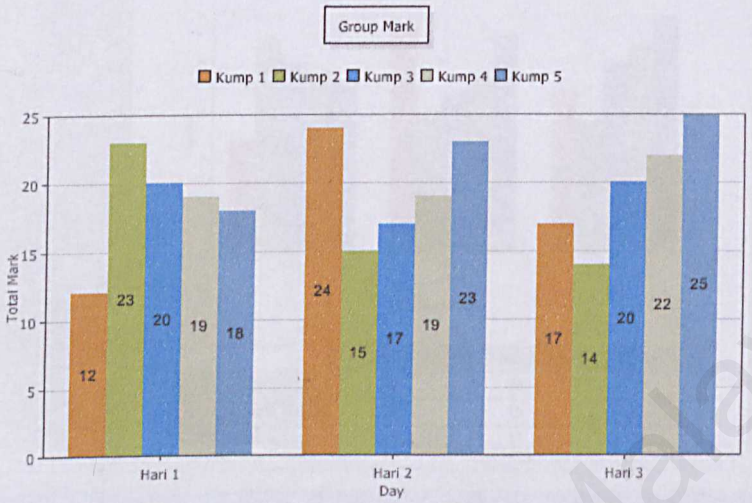


Table of Group

	Hari 1	Hari 2	Hari 3
Kump 1	12	24	17
Kump 2	23	15	14
Kump 3	20	17	20
Kump 4	19	19	22
Kump 5	18	23	25

List for total mark for every group by day

Analysis

Total	288
Maximum Data	25 (1 times at Hari 3, Kump 5)
Minimum Data	12 (1 times at Hari 1, Kump 1)
Average	19.2
Min	19.2
Medium	19
Mod (frequency)	

Figure 3.16: Example report layout with chart, table and analysis

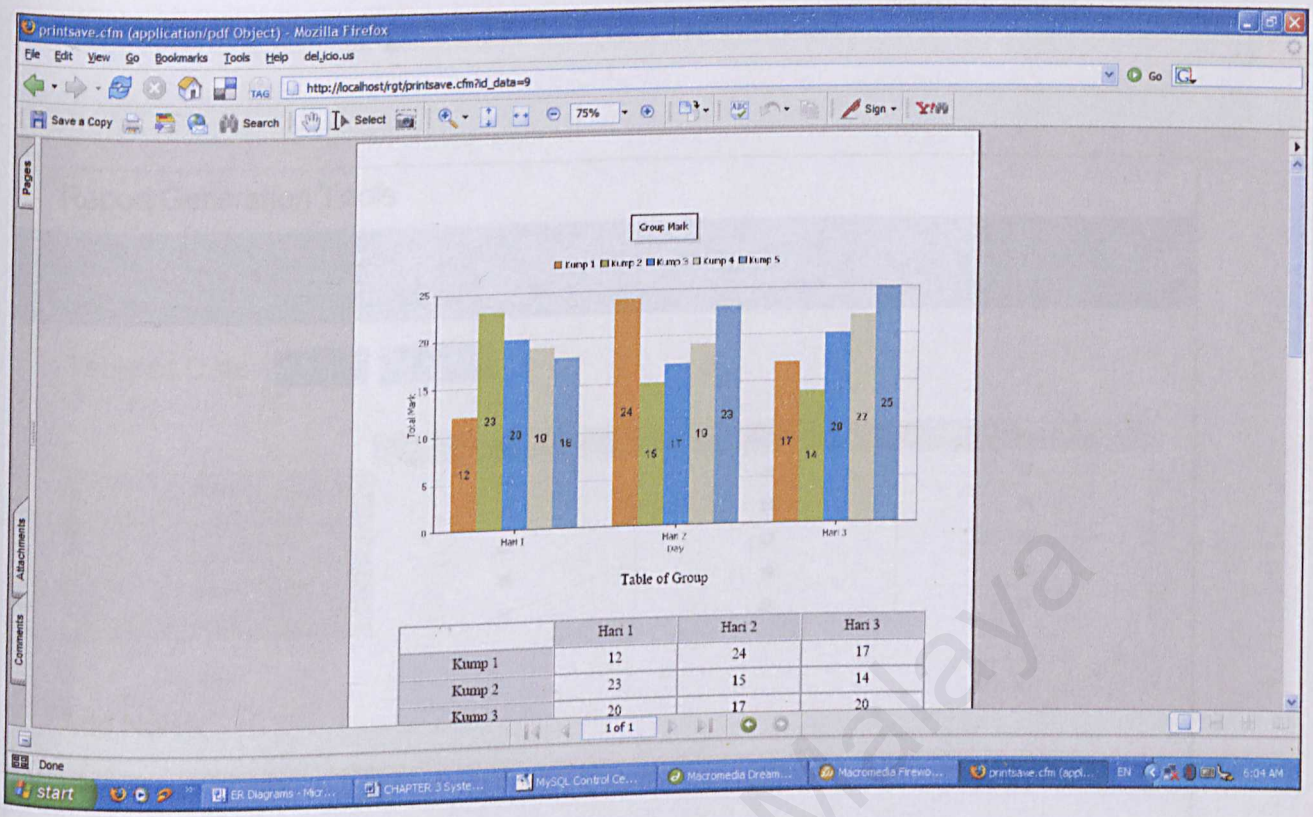


Figure 3.17: Print or Save View in PDF

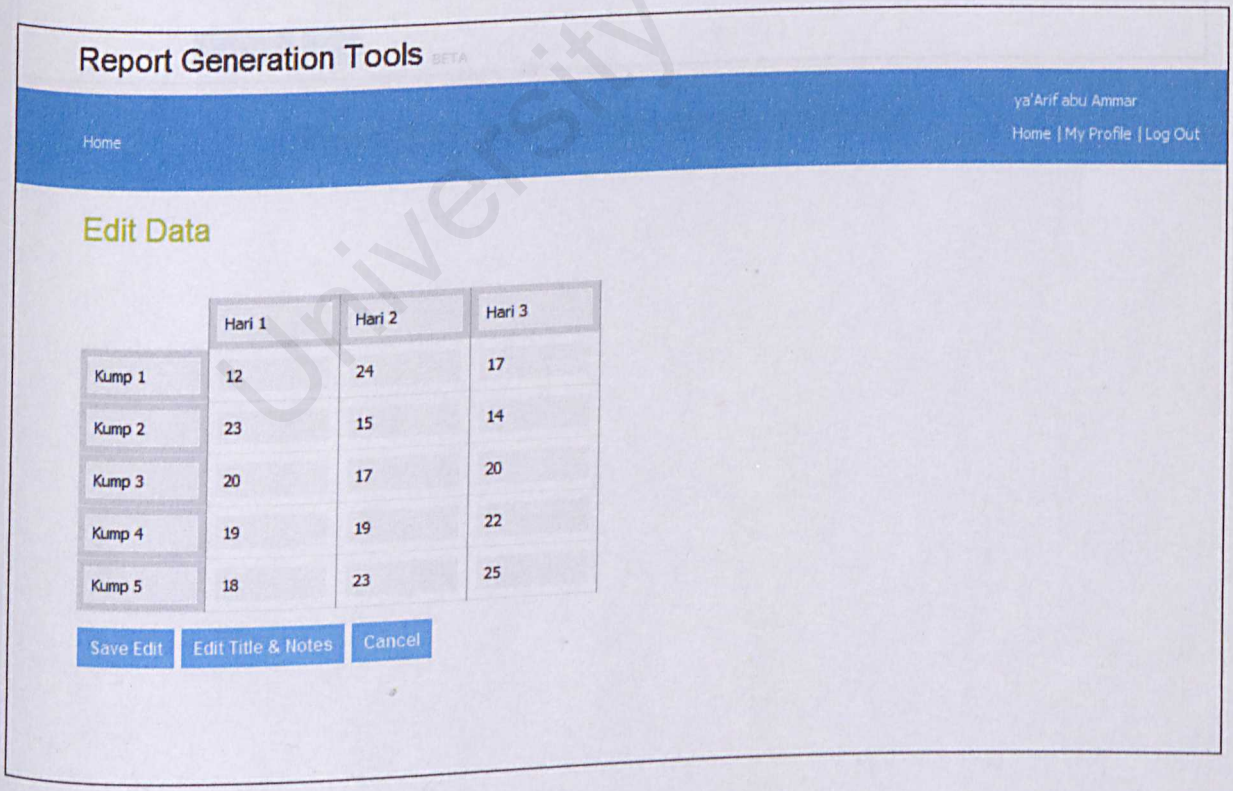


Figure 3.18: Interface for edit data.

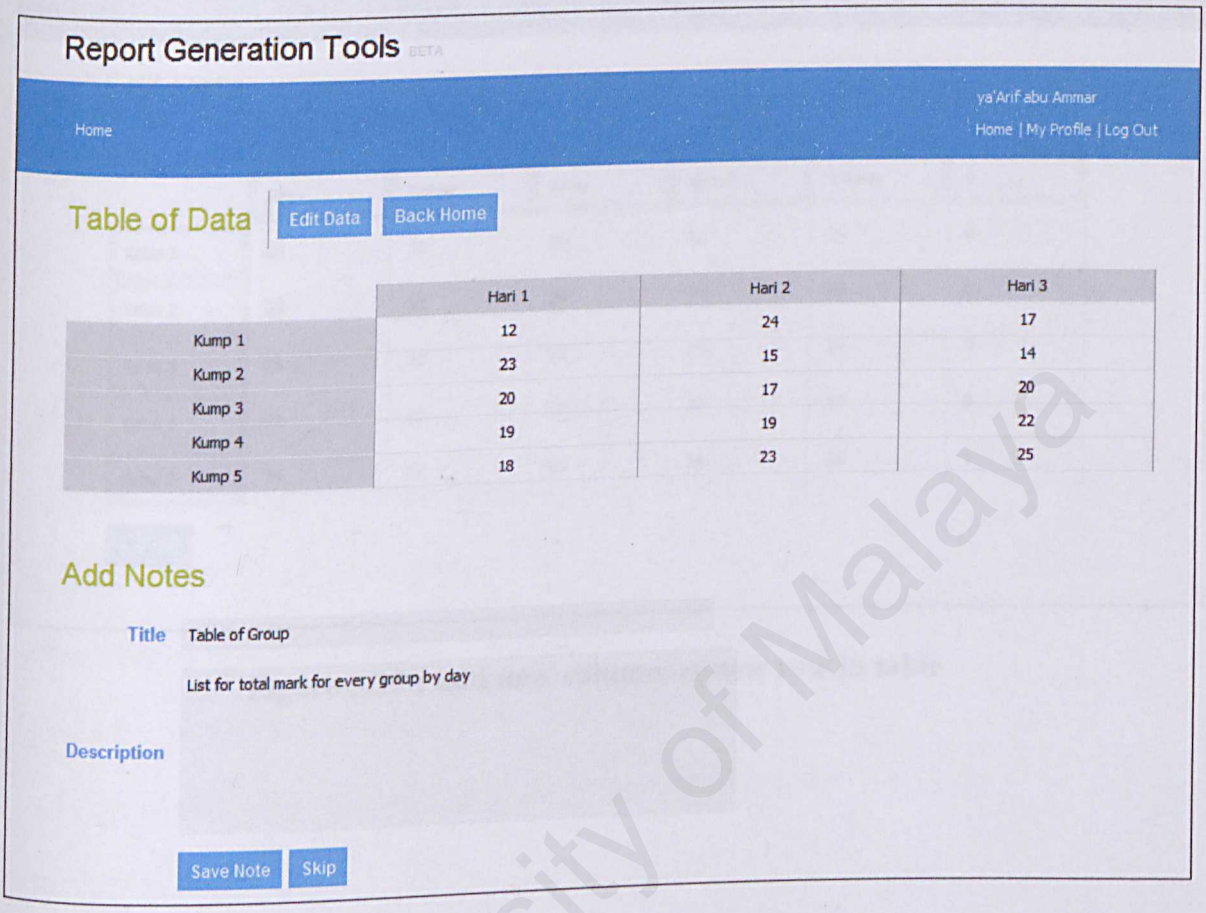


Figure 3.19: Interface to edit title and notes

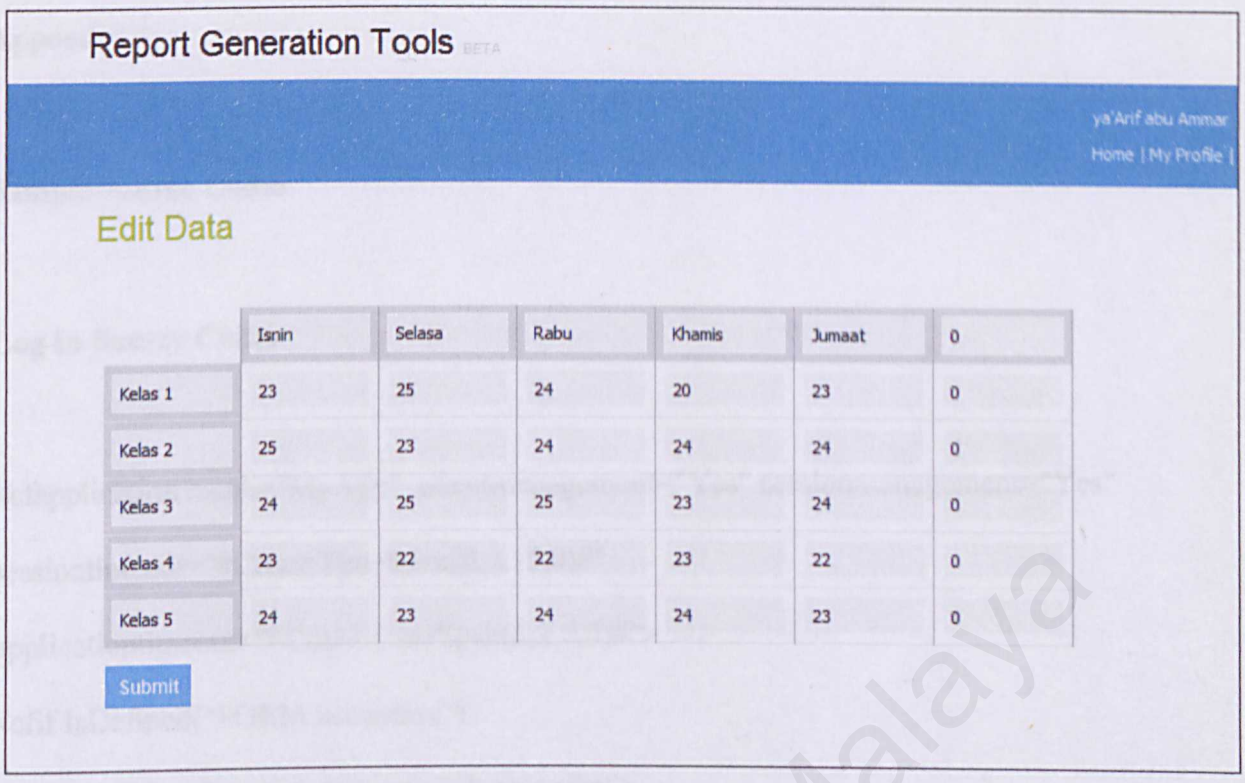


Figure 3.20 : add new column or new row in table

Appendix E

Sample Source Codes

Log In Source Codes

```
<cfapplication name="MyApp"  clientmanagement="Yes" sessionmanagement="Yes"
sessiontimeout="#CreateTimeSpan(0,0,15,0)#"
applicationtimeout="#CreateTimeSpan(0,2,0,0)#">
<cfif IsDefined("FORM.username")>
  <cfset MM_redirectLoginSuccess="main.cfm">
  <cfset MM_redirectLoginFailed="index.cfm?error=1">
  <cfquery  name="MM_rsUser" datasource="rgt_db" username="root"
password="isha04">
    SELECT username,password FROM user_db WHERE
username='#FORM.username#' AND password='#FORM.password#'
  </cfquery>
  <cfif MM_rsUser.RecordCount NEQ 0>
    <cftry>
    <cflock scope="Session" timeout="30" type="Exclusive">
      <cfset Session.MM_Username=FORM.username>
      <cfset Session.MM_UserAuthorization="">
    </cflock>
```

```
<cfif IsDefined("URL.accessdenied") AND true>
  <cfset MM_redirectLoginSuccess=URL.accessdenied>
</cfif>
<cflocation url="#MM_redirectLoginSuccess#" addtoken="no">
<cfcatch type="Lock"><!--- code for handling timeout of cflock --->
</cfcatch>
</cftry>
</cfif>
<cflocation url="#MM_redirectLoginFailed#" addtoken="no">
<cfelse>
<cfset MM_LoginAction=CGI.SCRIPT_NAME>
<cfif CGI.QUERY_STRING NEQ "">
  <cfset MM_LoginAction=MM_LoginAction & "?" &
XMLFormat(CGI.QUERY_STRING)>
</cfif>
</cfif>
```

Generate Chart Source Code

```
<tr>
        <td colspan="2" valign="middle">
<!--- Create chart --->
        <cfparam name="url.skin" default="default">
        <div align="center">

        <cfchart format= "#vformat#"
        showlegend="#interface_save.legend#"
        xaxistitle="#interface_save.label_x#"
        yaxistitle="#interface_save.label_y#"
        font="Tahoma" fontsize="11"
        title="#interface_save.title#"
        chartwidth="600"
        chartheight="400"
        seriesplacement="#interface_save.splace#"
        show3d="#interface_save.show3d#"
        showborder="no"
        pieslicestyle="solid"
        style="#url.skin#">
        <cfset count = 1 >
<cfset cp1 = #data_save.t_column# + 1>
```

```
<cfloop index="rgt_rowc" from="1" to="#data_save.t_row#">
<cfset count2 = ( rgt_rowc * cp1) + 1>
<cfchartseries datalabelstyle="value" type="#interface_save.chart_type#"
seriesLabel="#myarray[rgt_rowc * cp1]#">
<cfloop index="rgt_column" from="1" to="#data_save.t_column#">
<cfchartdata item="#myarray[rgt_column]#" value="#myarray[count2]#"/>
<cfset count2 = count2 + 1 />
 </cfloop>
</cfchartseries>
 <cfset count = count + 1 />
</cfloop>
</cfchart>
<!--- End Create Chart --->
</div></td>
</tr>
```

Generate Table Source Code

```
<tr>
<td colspan="2">
<cfif interface_save.data_table IS 1>
<table width="100%" border="0" cellspacing="0" cellpadding="5">
<tr>
<td><div align="center" class="tit"><cfoutput>#data_save.title#</cfoutput></div></td>
</tr>          <tr>
<td> </td>
</tr>
</table>
<table width="100%" border="0" cellspacing="0" cellpadding="5">
 <!--- Create table --->
          <cfset count = 1 />
       <cfset totalc = ArrayNew(2)>
       <cfset tc = 0 >
       <cfset tc2 = 0 >
          <cfloop index="rgt_row" from="0" to="#data_save.t_row#">
<tr> <cfloop index="rgt_column" from="0" to="#data_save.t_column#">
<cfif rgt_row IS 0 AND rgt_column IS 0 >
<td class="line_right_gray line_bot_gray"></td>
```

```
<cfelse>
<cfset totalc[rgt_row+1][rgt_column+1] = myarray[count] >
<!--- from array(1) to array(2) --->
<cfif rgt_row IS NOT 0>
<cfif rgt_column IS NOT 0 >
<cfset tc = tc + totalc[rgt_row+1][rgt_column+1] >
</cfif></cfif>
<td  <cfif #rgt_row# IS 0 OR #rgt_column# IS 0 >bgcolor="#CCCCCC"</cfif>
class="line_right_gray line_bot_gray"><div
align="center"><cfoutput>#myarray[count]#</cfoutput></div></td>
<cfset count = count + 1 />
</cfif>
</cfloop>
 </tr>
</cfloop>
<!--- End Create table --->
</table>
 </cfif> </td>
 </tr>
```

Analysis Source Code

```
<tr> <td colspan="2"> </td>
 </tr><cfif interface_save.id_anly IS NOT "">
<tr><td colspan="2" class="sub_tajuk">Analysis</td>
</tr> <tr>
<!--- Analysis Max and Min data --->
<cfset c = -1>
<cfset xmax = 0>
<cfset ymax = 0>
<cfset cmax = 1>
<cfset d = 100000000000000> <!--- Default value --->
<cfset xmin = 0>
<cfset ymin = 0>
<cfset cmin = 1>
<cfloop index="cc" from="0" to=#data_save.t_row# >
<cfloop index="dd" from="0" to=#data_save.t_column# >
<cfset ee = cc + 1 ><cfset ff = dd + 1 >
<cfif ee IS NOT 1 AND ff IS NOT 1>
<cfif c LTE totalc[ee][ff]>
<cfif c IS totalc[ee][ff]>
<cfset cmax = cmax + 1>
<cfelse>
```

<cfset cmax = 1>

</cfif>

<cfset xmax = ee><cfset ymax = ff>

<cfset c = totalc[ee][ff]>

<!--- Max --->

</cfif>

<cfif totalc[ee][ff] LTE d>

<cfif d IS totalc[ee][ff]>

<cfset cmin = cmin + 1>

<cfelse>

<cfset cmin = 1>

</cfif>

<cfset xmin = ee><cfset ymin = ff>

<cfset d = totalc[ee][ff]>

<!--- Min --->

</cfif>

</cfif>

</cfloop>

</cfloop>

<!--- Close Analysis Max and Min Data --->

<!--- Average/Min --->

<cfset ave = 0 >

<cfset ave = tc/(#data_save.t_row#*#data_save.t_column#)>

```
<!--- End Average --->

<!--- Medium --->

<cfset cmed = 0>

<cfset ff = 1>

<cfset coorm = 0>

<cfset newf = 0>

<cfset ctotal = ArrayNew(1)>

<cfset ctotal2 = ArrayNew(1)>

<cfset cm = #data_Save.t_column#+1>

<cfset cs = #data_save.t_column#+1>

<cfset rc = (((#data_save.t_column#+1)*(#data_save.t_row#+1))-1)>

<cfloop index="a1" from="#cs#" to="#rc#">

<cfset ctotal[ff] = myarray[a1]>

<cfset ff = ff +1 >

</cfloop>

<cfset ff = 1>

<cfloop index="a2" from="1" to="#ArrayLen(ctotal)#">

<cfif a2 MOD cm IS NOT 1>

<cfset ctotal2[ff] = ctotal[a2]>

<cfset ff = ff +1 >

</cfif>

</cfloop>

<cfset ArraySort(ctotal2,"numeric","asc") >
```

<cfset ta2 = ArrayLen(ctotal2) >

<cfif ArrayLen(ctotal2) MOD 2 IS 0>

<cfset cmed = (ctotal2[(ta2/2)]+ctotal2[((ta2/2)+1)])/2>

<cfelse><cfset newf = JavaCast("int", ((ta2/2)+1))>

<cfset cmed = ctotal2[newf]> </cfif>

<!--- End Medium --->

<!--- Mod

<cfset numbasing = ArrayNew(1)>

<cfset arraytemp = ArrayNew(2)>

<cfset countasing = 1>

<cfset numbasing[1] = ctotal2[1]>

<cfset creal =1>

<cfloop index="asing2" from="1" to="#ta2#">

<cfloop index="asing" from="1" to="#ta2#">

<cfif asing IS NOT ta2>

<cfif numbasing[1] is ctotal2[asing]>

<cfset countasing = countasing + 1 >

<cfelse>

<cfset arraytemp[creal][1]= numbasing[1]> <cfset arraytemp[creal][2]= countasing >

</cfif></cfif></cfloop><cfset creal = creal + 1><cfset numbasing[1] =

ctotal2[countasing+1]> <cfset countasing = 1></cfloop><cfset modnum = 0>

<cfset cmodnum = 0><cfset cmod = 0>

<cfloop index="cmod2" from="1" to="#creal#">

```
<cfif arraytemp[#cmod2#][2] GTE modnum >
<cfset cmod = arraytemp[cmod2][1]>
<cfset modnum = arraytemp[cmod2][2]>
</cfif>
</cfloop>
<!--- End Mod --->
<!--- view Analysis --->
<cfset myla = #interface_save.id_anly#>
<cfset myanaly=ArrayNew(1)>
<cfset myanaly = listtoarray(myla,",")>
<cfset arlen = ArrayLen(myanaly)>
<!--- End View Analysis --->
<td colspan="2"><table width="100%" border="0" cellspacing="0" cellpadding="5">
<tr> <td width="10%" nowrap="NOWRAP" class="line_bot_gray">Total </td>
<td width="90%" class="line_bot_gray">
<cfoutput></cfoutput><cfoutput>#tc#</cfoutput></td> </tr>
<cfloop index="analy_num" from="1" to="#arlen#" ><cfif myanaly[analy_num] IS 1 >
<tr><td nowrap="NOWRAP" class="line_bot_gray">Maximum Data</td>
 <td width="90%" class="line_bot_gray"><cfoutput>#c# ( #cmax# times
<cfif cmax IS 1> <cfoutput>at #totalc[1][ymax]#, #totalc[xmax][1]#</cfoutput>
</cfif> )</cfoutput></td>
</tr> <tr>
 <td nowrap="NOWRAP" class="line_bot_gray">Minimum Data </td>
```

```
<td class="line_bot_gray"><cfoutput>#d# ( #cmin# times <cfif cmin IS 1><cfoutput>at
#totalc[1][ymin]#, #totalc[xmin][1]#</cfoutput></cfif> )</cfoutput></td>
</tr></cfif><cfif myanaly[analy_num] is 2 >  <tr>
 <td nowrap="nowrap" class="line_bot_gray">Average</td>
<td class="line_bot_gray"><cfoutput>#ave#</cfoutput></td> </tr>
</cfif><cfif myanaly[analy_num] is 3 >
 <tr>
<td nowrap="nowrap" class="line_bot_gray">Min </td>
<td class="line_bot_gray"><cfoutput>#ave#</cfoutput></td>
</tr></cfif><cfif myanaly[analy_num] is 4 >
 <tr>
<td nowrap="nowrap" class="line_bot_gray">Medium  </td>
<td class="line_bot_gray"> <cfoutput>#cmed#</cfoutput></td>
</tr></cfif><cfif myanaly[analy_num] is 5 > <tr>
<td nowrap="nowrap" class="line_bot_gray">Mod (frequency) </td>
<td class="line_bot_gray"> </td> </tr>
</cfif></cfloop> </table></td>
 </tr></cfif>
```