ACCEPTANCE SHEET

The special problem entitled "*TERMS (Ticket Escalation and Resolution Management System) with Treemap-based reporting system*" prepared and submitted by *Melissa Lunasin* in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science has been examined and is recommended for acceptance.

_____

_

Richard Bryann L. Chua, M.Sc.
Adviser

EXAMINERS	APPROVED	DISAPPROVED
1. Gregorio B. Baes, Ph.D. (candidate)	_____	_____
2. Avegail D. Carpio, M.Sc.	_____	_____
3. Aldrich Colin K. Co., M.Sc. (candidate)	_____	_____
4. Vincent Peter C. Magboo, M.D., M.Sc.	_____	_____
5. Ma. Sheila A. Magboo, M.Sc.	_____	_____
6. Geoffrey A. Solano, M.Sc.	_____	_____

_____

Date

Accepted and approved as partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science

_____

_____

Geoffrey A. Solano, M.Sc.
Unit Head
Mathematical and Computing Sciences Unit
Department of Physical Sciences and Mathematics

Marcelina B. Lirazan, Ph.D.
Chair
Department of Physical Sciences and Mathematics

_____

Reynaldo H. Imperial, Ph.D.
Dean

College of Arts and Sciences

Abstract

The TERMS (Ticket Escalation and Resolution Management System) is a trouble-ticket system for storing and monitoring DSL-related complaints of PLDT DSL subscribers. Compared to the existing system, TERMS includes a treemap-based reporting module which summarizes all resolved DSL-related complaints stored into the system. Treemaps are space-filling figures used for visualizing large hierarchical data. Treemaps effectively use space, can display more information aside from structural relationship in one glance, can exploit color shading to represent attributes, and can help the user immediately identify large areas at once. In the treemap-based report, treemap properties such as the attribute to map color and size, duration covered and hierarchy of classification of nodes are customizable.

Keywords: trouble-ticket system, squarified treemap algorithm

Table of Contents

I. Introduction

A. Background of the Study

Philippine Long Distance Telephone Company (PLDT) is the leading telecommunications provider in the Philippines [1]. PLDT uses and maintains a wide range of equipment and devices such as routers and cables to provide service to their clients. It is inevitable that PLDT, a service-oriented company, will encounter technical problems during the operation of their equipment.

PLDT uses the Digital Subscriber Line (DSL) Subscriber Management System (DSMS) for the purpose of storing and searching through all the DSL-related complaints. It is accessible to personnel affiliated with the Help Desk Division, Business Zone Division, Fulfillment Division, Network Management Operations Support System (NMOSS) Division and the Network Resolution and Fault Escalation Division (NETRES) Division. All users can view all complaints input into the system. However, only the Help Desk Division, Business Zone Division, NMOSS Division and Fulfillment Division can modify the database of complaints. The NETRES Division personnel do not have modification privileges since it is not their responsibility to contact the clients to confirm if the problem has been solved, since client confirmation that a problem has been resolved is needed before closing a ticket [2].

The centralized Help Desk Division accepts calls from all DSL subscribers in the Philippines everyday. They are in charge of encoding the complaints into DSMS. All complaints are assigned a unique ticket number. The Help Desk employee would assign himself/herself as the first handler of the complaint. He/she would first try to identify the cause of the problem by inquiring about possible customer-premises related problems. If it was found to be a customer-premises related problem and it was solved, he/she could close the ticket. Closing a ticket requires indicating remarks on the nature of the problem and how it was solved. If the handler could not identify the problem, he/she could stage the ticket to the Business Zone Division whose station is nearest to the location of the complainant. Ticket staging is the process of transferring authority to handle a ticket to a predetermined, more specialized Division. In the case of the Help Desk Division, the Business Zone Division is the next more specialized Division. The Help Desk employee handling the ticket calls that particular Business Zone Division. Through private branch exchange, the idlest phone rings and anyone can answer the call and accept the ticket. The Help Desk employee records the complaint handler from the Business Zone Division and the Business Zone Division the complaint handler belongs to in the queue field [2].

The Business Zone Division member handling the ticket would then perform his/her checking procedures. In cases when Business Zone Division member could not attend to tickets assigned to him/her, he/she could endorse another member of his Division to take charge of the tickets staged to him/her. The assigned Business Zone Division member would then be the only one responsible for handling the ticket. The handler could close the ticket if he/she could identify and solve the problem and personally receive the confirmation from the client. If he/she cannot

solve the problem, he/she would contact the Fulfillment Division whose scope includes that area. Through private branch exchange, the idlest phone in the Fulfillment Division rings and anyone from that particular Fulfillment Division could answer the call and accept the ticket. The Business Zone Division member handling the ticket records the Fulfillment Division member who accepted the ticket as the complaint handler and Fulfillment Division the complaint handler belongs to in the queue field [2].

The Fulfillment Division member handling the ticket would check for node-level problems. If a Fulfillment Division member could not attend to his duties, he/she should endorse another member of his division to handle them for him/her. The Fulfillment Division deploys Business Zone Division/s to check equipment status and confirm from the client. The handler could close the ticket once he/she has identified and solved the problem and confirmed from the client that the problem no longer exists. One of the Fulfillment Division's tasks is to observe trends in the nature of the complaints, areas and nodes. If he deduces that there is a problem in the network, he contacts the NETRES division and reports his findings. Similar to the previous divisions, private branch exchange ensures that the idlest phone rings in the NETRES division. Anyone can answer the phone and work on the ticket. The indicated queue is NR1 for single node, NR2 for common area and NR3 for nationwide. A single node problem involves an access gateway (the first equipment accessed by subscribers) while a common area problem involves more sophisticated equipment such as switches and routers. If it was indeed a network problem, the Fulfillment Division will be contacted by NETRES personnel [2].

The NETRES division is in charge of checking network-level problems. Once solved, the NETRES employee in charge contacts the Fulfillment Division member handling the ticket and informs him/her about what the identified problem was and how it was dealt with. The handler from the Fulfillment Division verifies from the client that the problem has been resolved, before closing the ticket. The Fulfillment Division member confirms the resolution to the problem by calling all clients affected by the network problem [2].

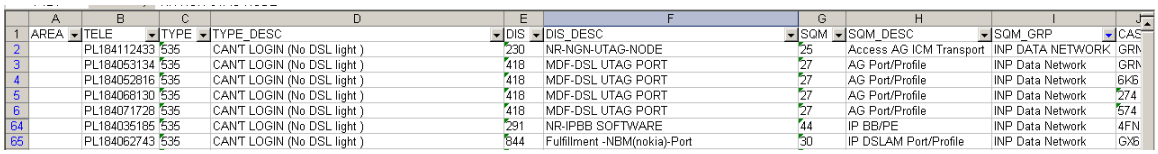
The NMOSS Division is responsible for specifying origin of the problem and responsible group to solve the problem(disposition - dis fields) and the cause and solution(cause - cas field) of each closed ticket from the remarks of the final handler of the ticket. One of the NMOSS Division's tasks is to produce daily reports summarizing the complaint database. These reports consist of 100% stack column charts showing the volume of complaints received classified by the general disposition, specific disposition and specific network element/segment involved in solving complaints. These reports are made for each region and each area in the Philippines [2].

Aside from checking for network-related problems, NETRES personnel are also tasked to produce reports as needed by NETRES division's head. On the average, these reports are submitted quarterly. They manually generate tables in MS Excel showing the tally of complaints for the duration specified by NETRES Division's head. The tables are classified by the type of node, trouble, general disposition, specific disposition, general network element/segment involved, specific network element/segment involved and the area. The report would be an important basis for improvement in technical and professional aspects. This comprehensive report of all complaints in the Philippines is submitted to the head of NETRES [2].

B. Statement of the Problem

PLDT's DSL Subscriber Management system has no report generating tool which would summarize the complaint-handling system. The NETRES and NMOSS are in charge of creating reports about the status of complaints in the Philippines and this is being performed manually. The complaints are exported to Excel format and the graphs are generated using MS Excel.

The system allows the NMOSS Division to directly export resolved DSL-related complaint records from the database of complaints in the form of a MS Excel file. This will serve as the data source for their daily report which will be further discussed. The NETRES Division member assigned to create the report requests from the NMOSS Division a copy of the complaint database containing resolved DSL-related complaints gathered within a certain period of time. The time interval depends on what the NETRES Division head specified and this is indicated by the NETRES personnel as he/she requests for the MS Excel files through email. A member of NMOSS Division attaches the MS Excel file when he/she replies to the requesting NETRES employee. The first row of the MS Excel file is composed of the field names. Each cell in the first row appears as a drop down box with options consisting of the unique values of that field in all the records in that MS Excel file. The rest of the rows contain the records. The following figure is an example of the exported database of complaints in MS Excel file.



	A	B	C	D	E	F	G	H	I	J
1	AREA	TELE	TYPE	TYPE_DESC	DIS	DIS_DESC	SQM	SQM_DESC	SQM_GRP	CAS
2		PL184112433	535	CAN'T LOGIN (No DSL light)	230	NR-NGN-UTAG-NODE	25	Access AG ICM Transport	INP DATA NETWORK	GRN
3		PL184053134	535	CAN'T LOGIN (No DSL light)	418	MDF-DSL UTAG PORT	27	AG Port/Profile	INP Data Network	GRN
4		PL184052816	535	CAN'T LOGIN (No DSL light)	418	MDF-DSL UTAG PORT	27	AG Port/Profile	INP Data Network	6K6
5		PL184068130	535	CAN'T LOGIN (No DSL light)	418	MDF-DSL UTAG PORT	27	AG Port/Profile	INP Data Network	274
6		PL184071728	535	CAN'T LOGIN (No DSL light)	418	MDF-DSL UTAG PORT	27	AG Port/Profile	INP Data Network	574
64		PL184035185	535	CAN'T LOGIN (No DSL light)	291	NR-IPBB SOFTWARE	44	IP BB/PE	INP Data Network	4FN
65		PL184062743	535	CAN'T LOGIN (No DSL light)	844	Fulfillment -NBM(nokia)-Port	30	IP DSLAM Port/Profile	INP Data Network	GX6

Figure 1. PLDT's exported complaint database

To create their report, the Daily DSL Service Quality Management (SQM) Analysis Report, the NMOSS Division uses the MS Excel file containing the resolved DSL-related tickets in the database for a particular day. PLDT divides the Philippine area into 5 regions, namely,

1. General Metro Manila North-East (GMMNE)
2. General Metro Manila South-West (GMMSW)
3. North Luzon
4. South Luzon
5. Visayas-Mindanao

Reports are generated using MS Excel. For each region, a graph summarizing all complaints is made, as well as graphs for each area belonging to that particular region. For each area, the total number of complaint records corresponding to each general disposition is counted and placed inside the MS Excel file near the actual graph. The tally consists of two columns, the left for the label of the general disposition and the right for the actual count for each. A 100% stacked column chart is created using the cells in the MS Excel file containing the label and tally for each general disposition as the data source. The generated graph is similar to a pie chart, but in the form of a vertical bar. On the left side of the chart, text boxes are placed to indicate labels and percentage each area in the chart represents. Beside the areas pertaining to a general disposition, each specific disposition is enumerated displaying their labels, the total count of each and their corresponding percentage relative to the other types of specific disposition. Below each specific

disposition, the specific network element/segment involved is also enumerated including their labels, the total count of each and their respective percentage. The values for the general disposition, specific disposition and specific network element/segment involved are derived by manually counting DSL-related closed tickets from the MS Excel file.

NETRES Division's report is produced in the same manner as with NMOSS Division's report. NETRES Division's report is comprised of tables showing the tally of complaints which fall under two or more criteria. For example, there is a table tallying the trouble type and the general disposition of the complaint for a particular type of node. Each row represents a trouble type and each column represents a disposition. Their intersection contains the count of complaints for the criteria based on the column and row to which it belongs to. Similar to NMOSS, NETRES personnel manually count the number of complaint records from the MS Excel file to get the values for the creation of the tables.

The system currently being used by PLDT, DSMS is mainly used as a repository of complaints and provides summaries of all tickets solved by each Business Zone Division member. The process of generating reports summarizing all the gathered DSL-related complaints in PLDT is currently being done manually by NETRES and NMOSS personnel. Since DSMS is a customized system purchased by PLDT from the company Netsoft, adding a module for the generation of reports would require contacting Netsoft again and the system might not be designed to allow for the smooth integration of a new module.

C. Objectives of the Study

To create the TERMS (Ticket Escalation and Resolution Management System) that allows:

1. Help Desk employee
    a. To create new ticket (complaint record)
    b. To stage a ticket to the Business Zone Division
    c. To close a ticket
    d. To browse all tickets
    e. To search all tickets
2. Business Zone employee
    a. To view the new tickets staged to him/her
    b. To stage a ticket to the Fulfillment Division
    c. To endorse a ticket to another Business Zone personnel to handle the ticket
    d. To close a ticket
    e. To browse all tickets
    f. To search all tickets
3. Fulfillment Division employee
    a. To view the new tickets staged to him/her
    b. To stage a ticket to the NETRES Division
    c. To endorse a ticket to another Fulfillment Division personnel to handle the ticket
    d. To close a ticket
    e. To browse all tickets

f. To search all tickets

4. NETRES Division employee

    a. To be view new tickets assigned to him/her

    b. To browse all tickets

    c. To search all tickets

    d. To generate a treemap of the distribution of complaints (daily by default) classified by the type of node on which it occurred, the reported trouble or nature of complaint, the general disposition, the specific disposition, the performed action and the area from which it occurred) with the following functions that could be performed on the treemap

        i. view the treemap

        ii. view the following details of a node (by left-clicking on the node)

            1. label
            2. number of nodes within the node
            3. the type of node
            4. the nature of complaint
            5. the general disposition
            6. the specific disposition
            7. the general network element/segment involved
            8. the specific network element/segment involved
            9. the general cause and solution
            10. the specific cause and solution
            11. the area

12. the number of complaints

13. the average length of time of service

iii. view a zoomed version of a node (by right-clicking on the node)

iv. rearrange the order of hierarchical classification of data

v. export the treemap into a JPEG file

vi. define the span of time covered by the treemap

vii. define the attribute of basis for the size of the nodes

viii. define the attribute of basis for the color of the nodes

ix. define the width of the padding between the subdivisions

x. define the size of the font to be used for labeling

xi. filter the nodes to be displayed

xii. save the defined settings

xiii. generate tabular format of the distribution of complaints

5. NMOSS Division employee

a. To indicate the origin of the problem and responsible group to solve the problem(disposition – dis field) and cause and solution(cause – cas field) to all complaint records

b. To browse all tickets

c. To search all tickets

d. To generate a treemap of the distribution of complaints daily classified by the type of node on which it occurred, the reported trouble or nature of complaint, the general disposition, the specific disposition, the performed action and the area

from which it occurred) with the following functions that could be performed on the treemap

i. view the resulting treemap

ii. view the following details of a node (by left-clicking on the node)

    1. label
    2. number of nodes within the node
    3. the region
    4. the area
    5. the general disposition
    6. the specific disposition
    7. the general network element/segment involved
    8. the specific network element/segment involved
    9. the general cause and solution
    10. the specific cause and solution
    11. the number of complaint records
    12. the average length of time of service

iii. view a zoomed version of a node (by right-clicking on the node)

iv. export the treemap into a JPEG file

v. define the width of the padding between the subdivisions

vi. define the size of the font to be used for labeling

vii. filter the nodes to be displayed

viii. save the defined settings

6. Administrator

a. Perform user management (add, edit and disable users)

b. Enable/disable HTTPS for login

D. Significance of the Study

DSL services are currently being availed not only by the private sector but also by companies. Since some companies depend on the internet for their business transactions and local users want to make the most out of their subscription, it is imperative for the service providers to assure them of continuous services. Substandard services may result to a bad reputation for the company. So companies like PLDT must ensure that they are able to solve complaints as soon as possible and personnel involved in complaint management are properly trained. In the long run, DSL subscribers would be more satisfied with their services and continue to patronize their services.

The proposed system would be of much use for the NETRES and NMOSS Division because they could eliminate the effort and time they spend in manually making reports from the whole database since the generation of graphs is a feature of the system. It would enable them to focus their attention on checking for the network problems.

Since the graph produced by the complaint system is in the form of a treemap, then it would be more comprehensive since the treemap will show all classifications of complaints encountered. The application will also be interactive and customizable so the NETRES and

NMOSS Division employee can adjust what the treemap displays depending on what trend he/she would be concerned about.

E. Scope and Limitations

- The application only covers complaints in DSL services.
- The application does not include an alarming system when a ticket is staged to a Division.
- The treemap application only includes complaint records which are completely filled up, or resolved tickets.

F. Assumptions

- The application assumes that complaints input into the system are only concerned with DSL services.
- Personnel are informed about new tickets assigned to him via telephone call from another team
- Personnel are informed about new tickets endorsed to him personally by the endorser.
- Customer details are retrieved from Integrated Customer Management System (ICMS), which is the accounts system of PLDT.

II. Review of Related Literature

Treemaps are space-filling figures designed to represent hierarchical structures [3]. Each node of the hierarchy is represented by rectangles having areas proportional to an attribute of the node such as node size [4]. This algorithm was conceptualized to aid in visualizing the hard disk of a computer. Nodes pertain to files and the interior nodes are the subdirectories [4]. Compared to other structures commonly used for visualizing hierarchical data, treemaps effectively use space, can display more information aside from structural relationship in one glance, can exploit color shading to represent attributes, and can help the user immediately identify large areas at once [4].

The original treemap algorithm, the Slice and Dice treemap algorithm, uses parallel lines to divide the area into smaller rectangles representing its children. For the succeeding hierarchy levels, the area is divided into alternating orientations (horizontal or vertical). The Slice and Dice treemap algorithm is simple to implement and changes made to the data would not have drastic effects on the drawn treemap [5]. In addition, the natural order of the nodes is preserved. However, the treemap produced tend to have very high aspect ratios [5]. Aspect ratio refers to the maximum between the height divided by the width and the width divided by the height. Rectangles with small aspect ratios tend to be square-like. As a result, comparing square-like rectangles and very thin rectangles becomes difficult [6]. Also, the skinny rectangles can be hard to view, select and label [5]. The Slice and Dice treemap algorithm is appropriate for nodes which are observed on a regular basis such as in stock portfolio monitors [5]. Figure 2 is an example of a Slice and Dice treemap.

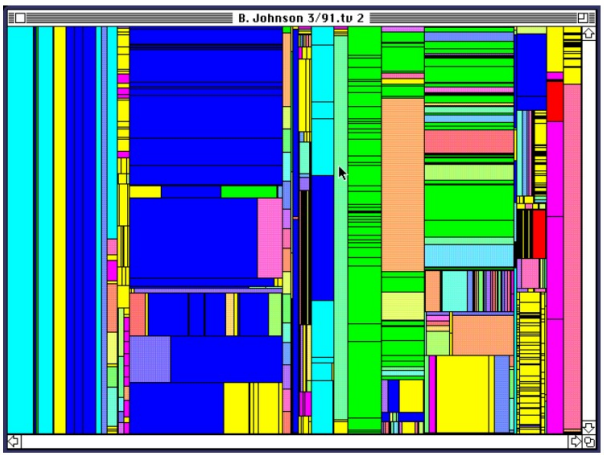


Figure 2. Slice and Dice Treemap

Ordered treemap algorithm was conceptualized to preserve order, producing layouts where items next to each other in a particular arrangement tend to be adjacent in the treemap [5]. Updating ordered treemaps causes the treemap to change relatively smoothly [5]. Although the produced aspect ratio is relatively small, it does not necessarily produce the best aspect ratios for the sets of rectangles. However, the ordered treemap algorithm requires the user to move his eye in four directions to actually observe the order [7]. Figure 3 is an example of an Ordered Treemap.

Figure 3. Ordered Treemap

Strip treemap algorithm produces rectangles in their natural order laid out in vertical or horizontal strips of different thicknesses. The next item in the list is added to the current strip and the dimensions of all rectangles in the strip are computed. If the average aspect ratio increased as a result of adding the new item, it is removed from the strip and a new strip is laid out. The ordering of nodes is preserved. The algorithm is efficient because it involves looking at rectangles within the current strip. Compared to ordered treemaps, strip treemaps are more readable. They also have similar aspect ratios and stability. However, strip treemaps entail the problem of laying out very thin final strips [5]. Figure 4 is an example of Strip treemap.

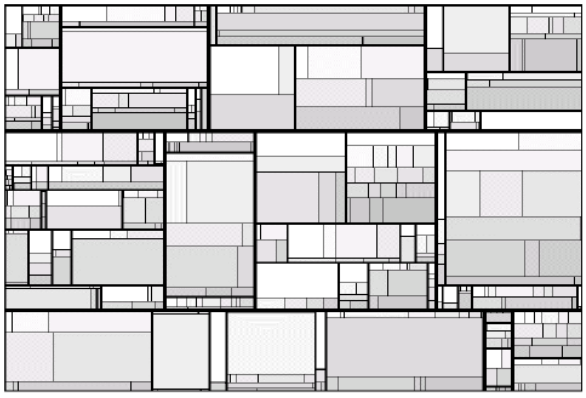


Figure 4. Strip Treemap

Quantum treemap algorithms are a group of algorithms extending the layout of existing treemap algorithms. The height and width of the produced rectangles are integer multiples of a specified elemental size [5]. This algorithm is appropriate for image browsers, where each image thumbnail is of a fixed size [5]. Compared to other treemap algorithms which focus on making rectangles more square-like, quantum treemap algorithms can produce layouts displaying images in regular sizes. Quantizing a treemap algorithm only adds a constant cost to the complexity of

the algorithm. However, quantizing treemap algorithms entails some problems. Example, in strip treemaps problems with areas with rough edges must be evened to produce rectangles for the treemap. Quantizing might produce problems in especially when applied to treemap algorithms that are not as straightforward as the strip treemaps [5]. Figure 5 is an example of a Quantum treemap.

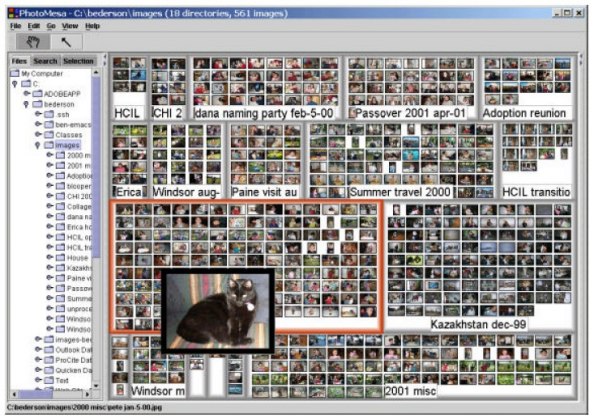


Figure 5. Quantum Treemap

The Cushion treemap algorithm was designed to improve visualization of the structure of nested treemap algorithms by making use of shadows. Instead of the usual margin along the sides of the children nodes of an area, shading of the child nodes allows the viewer to identify at once the structure. However, this algorithm requires additional computation to generate the shades for each rectangle. [8] Figure 6 shows an example of a Cushion treemap.

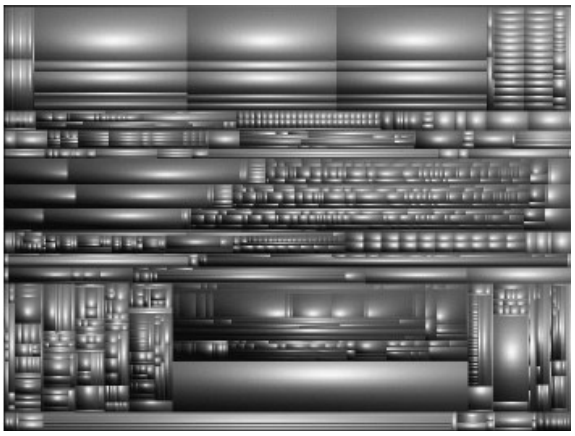


Figure 6. Cushion Treemap

Modifiable treemap algorithm was designed to make comparing of rectangles' sizes easier. The user is allowed to specify the aspect ratio of rectangles, resulting to rectangles with similar shapes, making them easier to compare. However, slight modifications in the sizes can cause large changes in the generated treemap and in the presence of many subnodes, the obtained layout contains figures close to the fixed ratio. [9] Figure 7 shows an example of a Modifiable treemap.

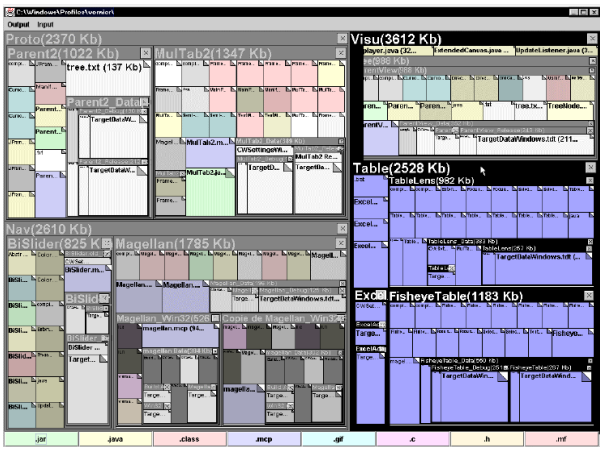


Figure 7. Modifiable Treemap

Voronoi treemap algorithm enables polygon-based presentation of treemaps. This algorithm was designed to generate low aspect ratios and improve readability and flexibility. The main drawback of this algorithm is the huge amount of computations necessary to generate the treemap. [10] Figure 8 shows an example of a Voronoi treemap.

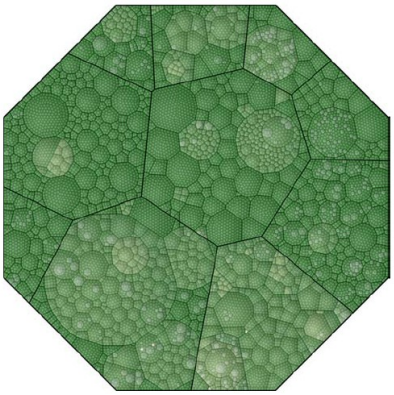


Figure 8. Voronoi Treemap

Cascaded Treemap Algorithm was designed to improve readability of the hierarchy of the treemap as well as conserve space. This algorithm is an improvement to nested treemaps, but each level is overlapped with the previous subdivision to emphasize the hierarchy. [11] Figure 9 shows an example of a Cascaded treemap.

Figure 9. Cascaded Treemap

Three-dimensional Treemap Algorithm was designed to improve visibility of the treemap by portraying the treemap in 3d space. Each sublevel of the hierarchy is stacked on top of its parent. However this method causes a slowdown in the treemap generation. [12] Figure 10 shows an example of a 3-d treemap.

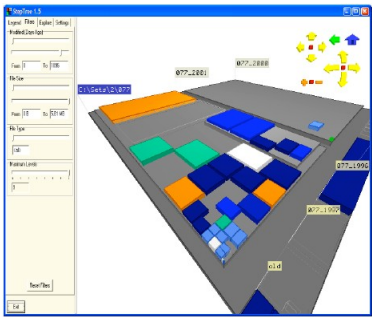


Figure 10. 3-d Treemap

The Squarified treemap algorithm was designed to produce rectangles with low aspect ratios; thus they are more square-like. Square-like rectangles would result to increased efficiency of the displayed space, visibility, comparability and accuracy of the presentation. However, the hierarchy becomes less noticeable, the order of nodes is lost and the images become less regular [13]. Figure 11 is an example of a Squarified Treemap.

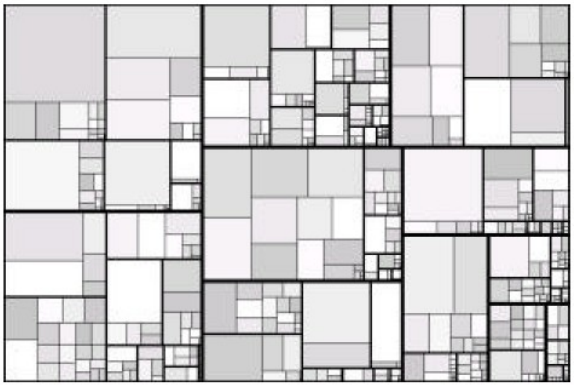


Figure 11. Squarified Treemap

III. Theoretical Framework

A. TERMS DSL-Complaint Resolution Protocol

TERMS currently has 2 tables: the ticket table and tracking table. The ticket table contains all complaints filed to PLDT. The tracking table is used to allow other uses to know who is handling a ticket at a certain point in time [2].

A ticket record has the following fields:

- TICKET_ID
- TELE – telephone number of the complainant
- TYPE – id of the nature of complaint/trouble
- DIS – the id of the disposition to the problem
- SQM – id of the specific network element involved
- SQM_GRP – id of the general network element involved
- CAS – id of the general cause and solution to the problem
- SUBCAS – id of the specific cause and solution to the problem
- REP_D – reported date
- REP_T – reported time
- AGE – duration in hours when the ticket was handled
- NAME – subscriber's name

- ADDR1 – subscriber's address
- AREA – the area covering the subscriber's location
- CTR2 – the region covering the subscriber's location
- CTR1 – the sub-region covering the subscriber's location
- MOFC – the main office covering the subscriber's location
- OFC – the office covering the subscriber's location
- SA –
- WA –
- CS –
- STRIP –
- PAIR –
- DP –
- PRIO –
- STN –
- NODE – the network node/access gateway id

A tracking record has the following fields:

- TRACKING_ID
- TICKET_ID
- TECH_ID – personnel handling the ticket
- TIME – date and time the employee started handling the ticket
- QUEUE – This has 2 meanings based on the level of the tracking record.

- division the handler belongs to (if the tracking record is at the Help Desk, Business Zone or Fulfillment Division level)
- type of network problem (if the tracking record is at the NETRES Division level)

- REMARKS

When a customer has a DSL problem with PLDT, he/she calls 171 and a Help Desk Division member takes the complaint. Help Desk Division members are the only users of the system who can add complaint records into the system. The Help Desk employee fills up a form for recording the complaint into the system. He/she has to input the TYPE_DESC and TELE fields for the complaint. A complaint record is added into the system. The system automatically fills in the TICKET_ID, TYPE, REP_D and REPT_T. The system automatically adds into the complaint record the following fields from the ICMS: NAME, ADDR1, AREA, CTR2, CTR1, MOFC, OFC, SA, WA, CS, STRIP, PAIR, DP, PRIO, STN and NODE [2].

A tracking record is created as complaint records are input into the system. The system automatically fills in the TRACKING_ID, TICKET_ID, TECH_ID, TIME and QUEUE (value is 171) [2].

Only the Help Desk Division member who encoded the complaint into the system can close the ticket. The Help Desk Division checks for customer-premises related problems. He/she closes the ticket if he/she can solve the problem. He/she then fills up a form for closing the ticket. The REMARKS field and SQM field are filled in by the Help Desk Division employee for

the tracking record. At the same time, the system automatically fills in the AGE field for the complaint record [2].

In cases when the handler from the Help Desk Division could not attend to his/her duties, he/she endorses another Help Desk Division member to take charge of the handler's tickets. The handler of the ticket fills in a form for endorsing another Help Desk Division Member. He/she fills in the REMARKS field for the current tracking record [2].

A tracking record is created for the next handler of the ticket, the Help Desk Division member endorsed by the current handler of the ticket. The Help Desk employee fills in the TECH_ID field for the new tracking record. At the same time, the system automatically fills in the TRACKING_ID, TICKET_ID, TIME and QUEUE (value is 171) fields for the new tracking record [2].

If the handler cannot solve the ticket, he/she stages the ticket to a member of the Business Zone Division whose post is nearest to the location of the complainant. Through private branch exchange, the idlest phone from that particular Business Zone Division rings and anyone can answer the call. The Business Zone Division is in charge of physically visiting equipment handled by PLDT to perform checking for some customer-premises related and port-related problems. Business Zone Divisions are assigned in areas such as Makati and Mandaluyong. Each Business Zone Division handles areas with approximately 1,000 km of radii in their assigned area. The Help Desk employee handling the ticket fills up a form for staging the ticket. He/she fills in the REMARKS field for the current tracking record [2].

A tracking record is created for the next handler of the ticket, the Business Zone Division member who accepted the ticket over the phone. The Help Desk employee fills in the TECH_ID field for the new tracking record. Also, the system automatically fills in the TRACKING_ID, TICKET_ID, TIME and QUEUE (the Business Zone Division of the next handler) fields for the new tracking record [2].

Only the Business Zone Division member assigned to a particular ticket can close it. If the handler solves the problem, he/she closes the ticket. He/she then fills a form for closing a ticket. The REMARKS field and SQM field are filled up by the handler for the tracking record. At the same time, the system automatically fills up the AGE field in the complaint record [2].

In cases when the handler from the Business Zone Division could not attend to his/her duties, he/she endorses another Business Zone Division member to take charge of the handler's tickets. The handler of the ticket fills in a form for endorsing another Business Zone Division Member. He/she fills in the REMARKS field in the current tracking record [2].

A tracking record is created for the next handler of the ticket, the Business Zone Division member endorsed by the current handler of the ticket. The Business Zone Division employee fills in the TECH_ID field for the new tracking record. At the same time, the system automatically fills in the TRACKING_ID, TICKET_ID, TIME and QUEUE (the Business Zone Division of the next handler) fields for the new tracking record [2].

If the handler cannot solve the ticket, he/she stages the ticket to a member of the Fulfillment Division whose scope includes the area of the business zone. Fulfillment Divisions are in charge of checking for node-level problems. Using Network Management Systems (NMS), they could monitor status of nodes under their jurisdiction. There are two Fulfillment Divisions, one for General Metro Manila (GMM) and another for North Luzon, South Luzon, Visayas and Mindanao. Through private branch exchange, the idlest phone from that particular Fulfillment Division rings and anyone can answer the call. The Business Zone Division employee handling the ticket fills up a form for staging the ticket. He/she fills in the REMARKS field for the old tracking record [2].

A tracking record is created for the next handler of the ticket, the Fulfillment Division member who accepted the ticket over the phone. The Business Zone employee fills in the TECH_ID field for the new tracking record. Also, the system automatically fills in the TRACKING_ID, TICKET_ID, TIME and QUEUE (the Fulfillment Division of the next handler) fields for the new tracking record [2].

Only the Fulfillment Division member assigned to a particular ticket can close it. If the handler solves the problem, he/she closes the ticket. He/she then fills a form for closing a ticket. The REMARKS field and SQM field are filled up by the handler for the tracking record. At the same time, the system automatically fills up the AGE field in the complaint record [2].

In cases when the handler from the Fulfillment Division could not attend to his/her duties, he/she endorses another Fulfillment Division member to take charge of the his/her tickets.

The handler of the ticket fills in a form for endorsing another Fulfillment Division Member. He/she fills in the REMARKS field for the current tracking record [2].

A tracking record is created for the next handler of the ticket, the Fulfillment Division member endorsed by the current handler of the ticket. The Fulfillment Division employee fills in the TECH_ID field for the new tracking record. At the same time, the system automatically fills in the TRACKING_ID, TICKET_ID, TIME and QUEUE (the Fulfillment Division of the next handler) fields for the new tracking record [2].

If the handler cannot solve the ticket, he/she stages the ticket to NETRES. Through private branch exchange, the idlest phone from NETRES rings and anyone can answer the call. The Fulfillment Division employee handling the ticket fills up a form for staging the ticket. He/she fills in the REMARKS field for the tracking record [2].

A tracking record is created for the NETRES Division. The Fulfillment Division employee fills in the QUEUE field for the new tracking record. The value of the QUEUE NR1 is for single node, NR2 for common area and NR3 for nationwide. Also, the system automatically fills in the TECH_ID (the id of the Fulfillment Division employee currently handling the ticket), TRACKING_ID, TICKET_ID and TIME fields for the new tracking record [2].

After the NETRES Division members solved the network problem, an assigned NETRES Division employee contacts the Fulfillment Division member formerly handling the ticket. The NETRES Division member provides the handler from the Fulfillment Division the remarks for

the resolved problem. The handler from the Fulfillment Division then fills up a form for closing that particular ticket staged to the NETRES Division. He/she fills up the REMARKS field and SQM field for the tracking record. At the same time, the system automatically fills up the AGE field in the complaint record [2].

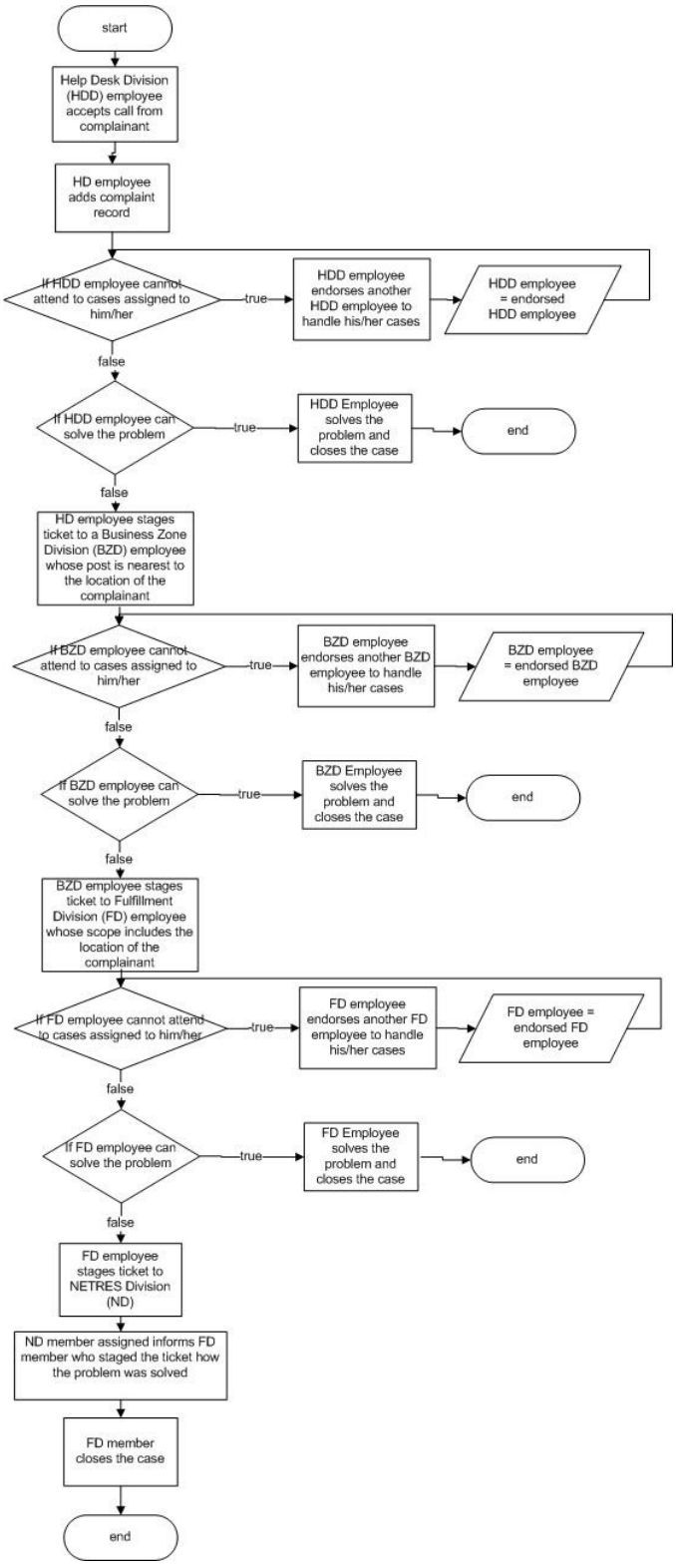


Figure 12. Flowchart of TERMS DSL-Complaint Resolution Protocol

B. Issue Tracking Systems

Issue tracking systems, also known as trouble ticket systems or incident ticket systems are computer applications used by organizations to store their issues addressed to them by their clients. These systems are commonly used in conjunction with customer support call centers for adding, updating and resolving complaints directed to companies. Issue tracking systems include knowledge bases or Database Management Systems (DBMS) for storing customer information, complaint details and other necessary data [14].

A DBMS is a program which allows users to save, edit, and retrieve information from a database. Queries are used to perform these operations on the content of the database [15].

C. Squarified Treemap Algorithm

The Squarified Treemap algorithm was independently conceptualized by Martin Wattenberg and the following people from Eindhoven University of Technology-Mark Bruls, Kees Huizing and Jarke J. van Wijk. The algorithm aims to produce treemap rectangles which are more square-like. The following are the advantages of square-like rectangles: [13]

1. Square-rectangles utilize space more efficiently.
2. The number of pixels for the border is relative to the total area of the rectangle. And squares are the only figures which could produce minimal borders.

3. They are easier to find and select; thin rectangles are messy and result in aliasing errors.
4. Since squarified treemaps produce rectangles which are nearly square-like, then they tend to have the same aspect ratios, as close to 1 as possible. Rectangles with similar aspect ratios are much easier to compare in size. [13]

The following are the steps for creating a squarified treemap:

1. Select dimensions of the treemap to be drawn.
2. Sort the values to be used as basis for the area of the rectangular nodes in descending order.
3. Scale the values such that they would fit the defined dimension of the treemap.
4. Split the initial rectangle. If the length is greater than the width, the division is horizontal. Otherwise, the division is vertical.
5. Lay out the first rectangle into the division.
6. Take note of the aspect ratio of the rectangle. Aspect ratio is the maximum between the length divided by the width and the width divided by the length.
7. Include another rectangle into the division whose area will be the next value from the sorted list. If the division is vertical, the laid rectangle is placed on top of the previous rectangle. If the division is horizontal, the laid rectangle is placed on the right of the previous one.
8. Get the aspect ratio of the added rectangle. If the aspect ratio is larger than the aspect ratio of the previously added rectangle, then do not include it in the current division.

9. Get the remaining rectangle from the whole area. The same procedures are performed until all values had been laid out. [13]

The following is the pseudocode of the squarified treemap algorithm:

```
Procedure squarify(list of children, list of row, real w)
begin
	real c = head(children);
	if worst(row,w) ≤ worst(row++[10],w) then
		squarify(tail(children),row++[10],w);
	else
		layoutrow(row);
		squarify(children,[],width());
	fi
end
```

where:

squarify function – procedure lays out the rectangles in horizontal and vertical rows

Rectangle – data type assumed which contains the layout of the children in one rectangle

width() – gives the length of the shortest side of the remaining subrectangle in which the current row is placed

layoutrow() – adds a new row of children to the subrectangle

++ – used to denote concatenation of lists

[x] – is the list containing element x

[] – is the empty list

worst – gives the highest aspect ratio of a list of rectangles, given the length of the side along which they are to be laid out [13]

The following is a simple example of how the squarified treemap algorithm works:

Suppose that the chosen rectangle where the treemap will be drawn has a length of 6 and a height of 4. The following values correspond to the areas which must be laid out in the main rectangle – 6, 6, 4, 3, 2, 2 and 1. Since the values are already in descending order, there is no need to sort them. Since the sum of the values to be used for the areas and the total area produced by the chosen dimensions for the treemap are the same, there is no need to scale the values.

First, the main rectangle is split. The division is horizontal because the length is longer than the height. The left rectangle is filled in with the first rectangle with an area of 6, the first value in the sorted list of areas. This rectangle has a computed aspect ratio is 8/3. The next rectangle whose area is the next value in the sorted list, 6 is laid on top of the previous one. The aspect ratio of the next rectangle is 3/2, which is smaller than the preceding rectangle. So the next rectangle with an area of 4, the third value in the sorted list is laid on top of the previous rectangle. The aspect ratio of the added rectangle becomes 4/1. Since 4/1 is larger than 3/2, the rectangle with an area of 4 is removed from the current division.

The current rectangle is assigned to be the remaining rectangle if first division containing the values 6 and 6 is removed. This area has a length of 3 and a width of 4. The division is vertical since the current rectangle's width is larger than the length. The next rectangle to be laid

out has an area of 4, the next value in the sorted list. The aspect ratio computed is 9/4. The next rectangle with an area of 3, the next value in the list is laid out at the right of the first rectangle. The aspect ratio computed is 49/27. Since 49/27 is smaller than 9/4, we continue adding rectangles to the current division. The next rectangle with an area of 2, the next value in the sorted list is laid out to the right of the previous rectangle. The computed aspect ratio is 9/2, which is larger than 49/27. Therefore, the rectangle with an area of 2 is removed from the current division.

The current area becomes the original rectangle when the 2 subdivisions are removed. This rectangle has a length of 3 and a height of 5/3. Since the length is larger than the width, the subdivision is horizontal. We first lay out the rectangle with an area of 2, the next item in the sorted list. The computed aspect ratio becomes 25/18. The next rectangle with an area of 2, the next value in the sorted list, is laid out on top of the previous one. The aspect ratio of the added rectangle becomes 144/50. Since 144/50 is larger than 25/18, the added rectangle with an area of 2 is removed from the current subdivision.

The remaining rectangle when all the subdivisions are removed becomes the current area. This area has a length of 9/5 and a width of 5/3. The first rectangle to be laid out has an area of 2, the next value in the sorted list. Since the next value, is the last value in the sorted list we simply place it in the remaining area.

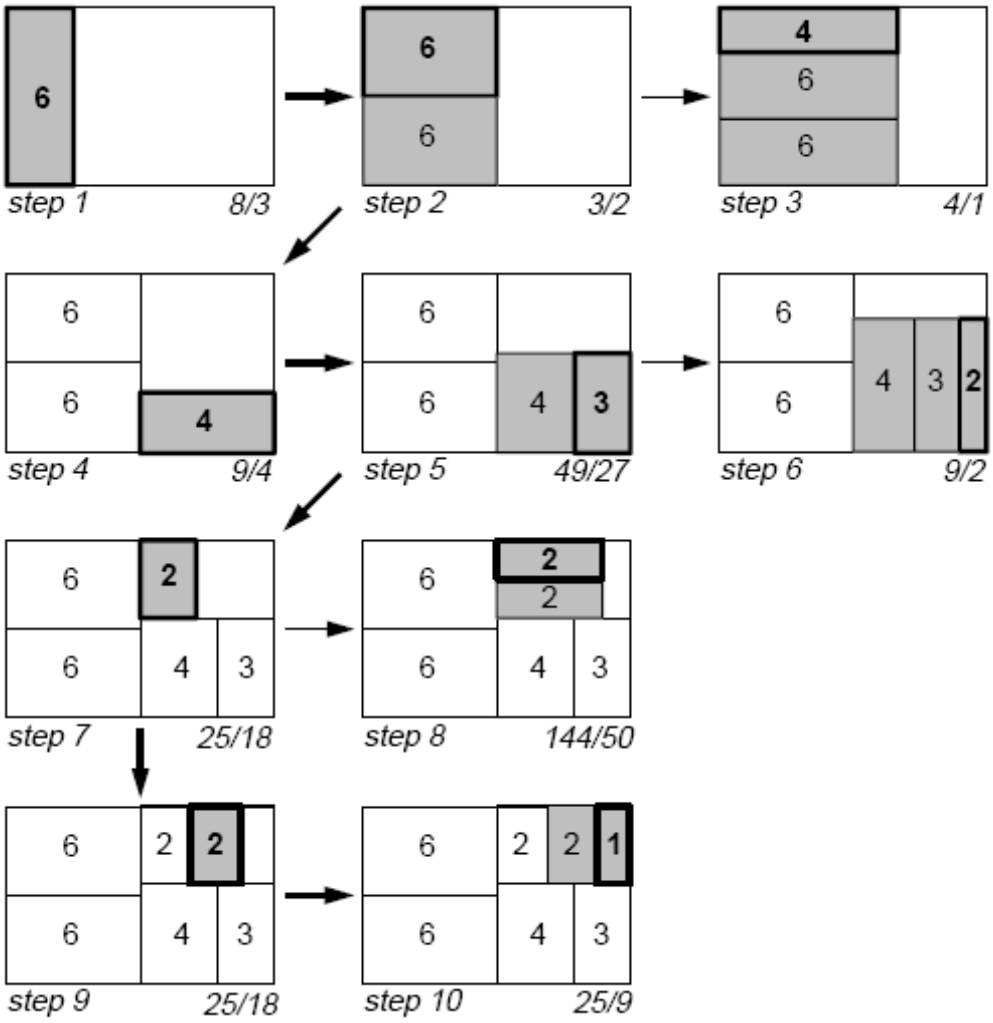


Figure 13. Example of constructing squarified treemap

C. Scalable Vector Graphics (SVG)

SVG or Scalable Vector Graphics is a platform for graphing 2-d images. It is mainly composed of two parts, an XML-based file format and a programming API for applications involving graphics. SVG features include figures, text and embedded raster graphics with various

rendering styles. SVG also supports scripting languages such as ECMAScript and animation. [16]

SVG is utilized in fields such as web graphics, animation, user interfaces, graphics interchange, print generation, mobile applications and high-grade design. SVG builds upon various standards including XML, JPEG and PNG for image formats, DOM for scripting, SMIL for animation and CSS for styling. [16]

IV. Design and Implementation

A. Context Diagram

The Ticket Escalation and Resolution Management System interacts with 5 types of users namely the Help Desk Division Employee, Equipment Specialist, NETRES Division Employee, the NMOSS Division Employee, the Administrator as well as another system, the ICMS. The Equipment specialist has two types, the Business Zone Division member and the Fulfillment Division member. The Context Diagram is shown in Figure 14.

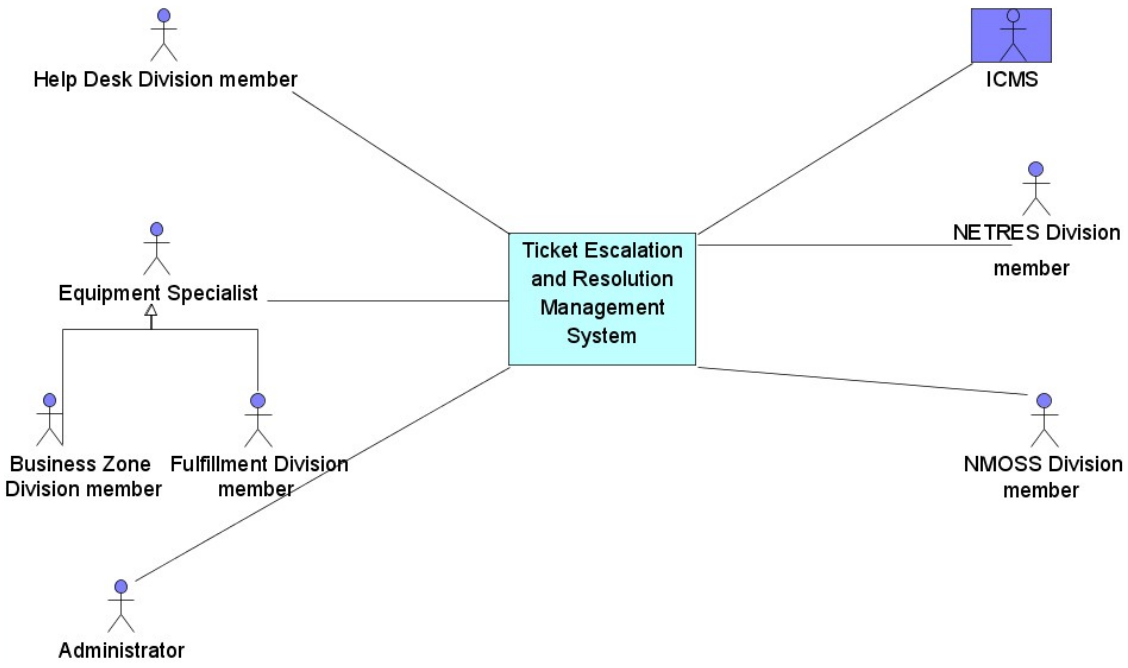


Figure 14. Context Diagram of TERMS

B. Use Case Diagram

Help Desk Division Members can record complaints, view complaint records, close tickets, endorse ticket and stage tickets. The complaint recording process involves retrieving information from the ICMS. Equipment Specialists can view tickets, close tickets, endorse members and stage tickets. NETRES Division members can view tickets and generate reports. NMOSS Division Members can view tickets, edit tickets and generate reports. The Administrator can manage users and enable or disable HTTPS for login. Figure 15 shows the top level use case diagram.

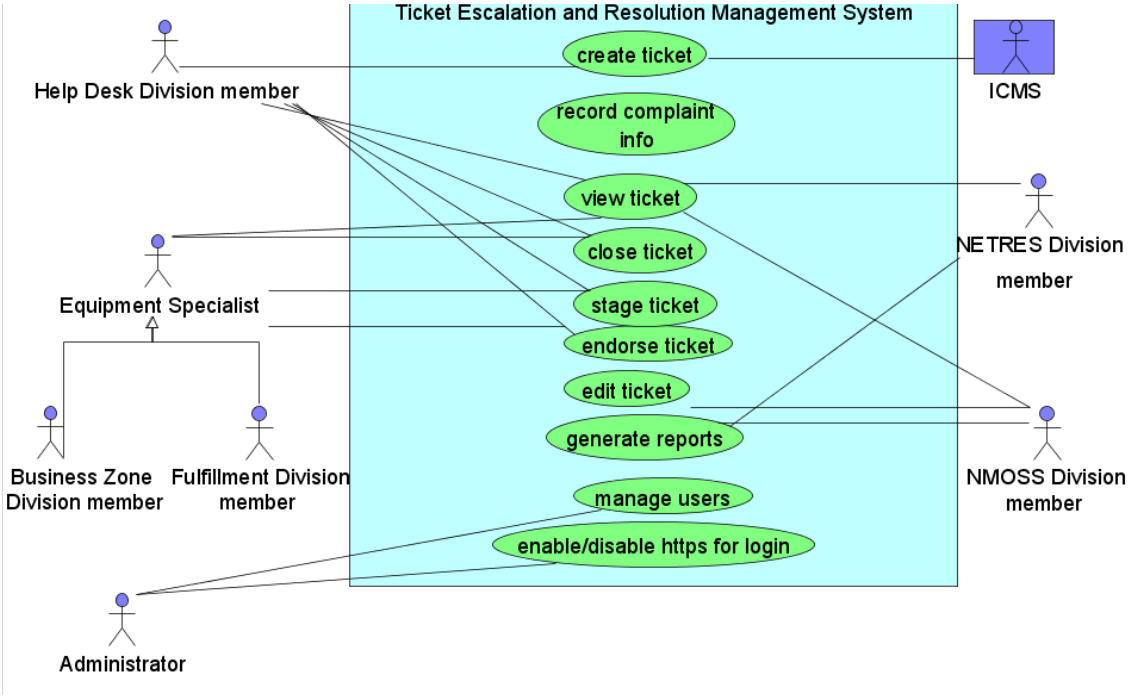


Figure 15. Top Level Use Case Diagram of TERMS

Create ticket use case includes verifying the subscriber and adding the complaint. Adding the complaint includes automatically generating ticket id, recording complaint information, recording subscriber information and assigning handler. Staging a ticket includes assigning handler. Endorsing a ticket also includes assigning handler. Generating reports use case includes displaying the report. Manage users use case includes adding, editing and disabling users. The detailed use case diagram is shown in Figure 16.

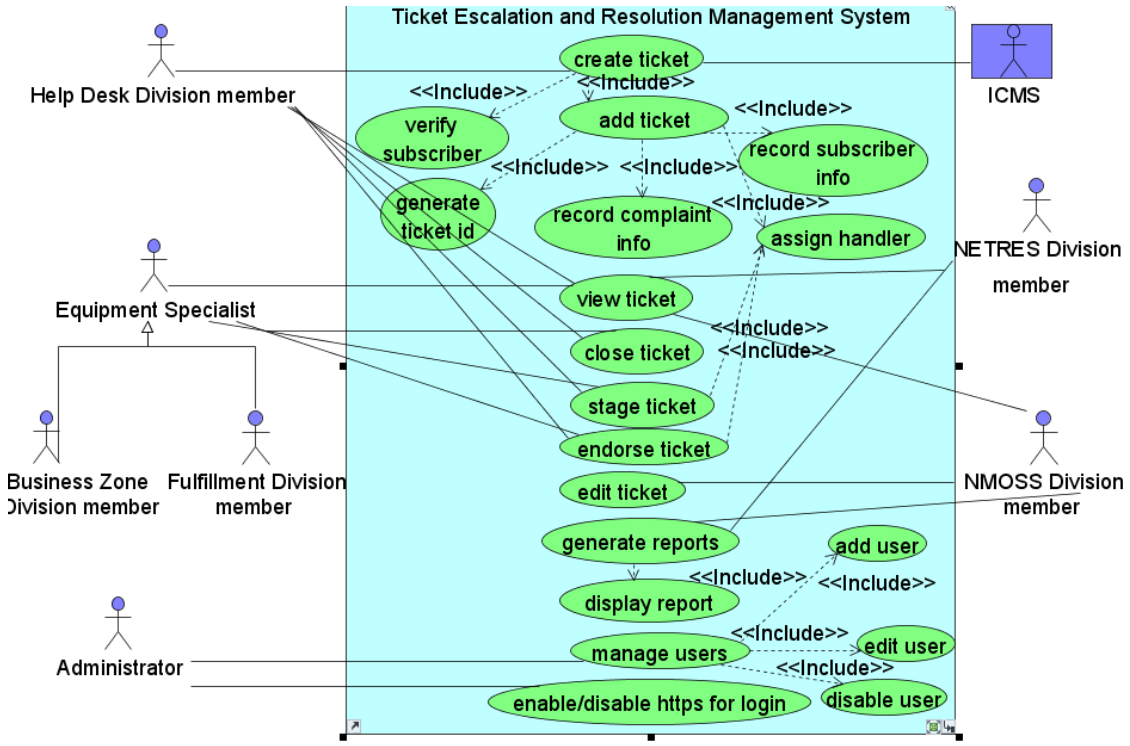


Figure 16. Detailed Use Case Diagram of TERMS

1. Create Ticket

The Create Ticket Use Case involves only the Help Desk Division Member, ICMS and TERMS. Recording the complaint involves verifying if the caller is a valid subscriber and adding the complaint. The process of verifying the subscriber is done by accessing the ICMS, the system for storing subscriber information. Adding the complaint includes automatically generating a ticket id, recording complaint information, recording subscriber information and assigning the handler. A unique ticket id is assigned to each ticket. Complaint information such as nature of the complaint and date and time when the complaint was filed are stored. Subscriber information which is stored in the ICMS is retrieved and added to the complaint record. The handler is assigned by the system to be the Help Desk Division member who created the ticket. Figure 17 shows the Create Ticket Use Case.

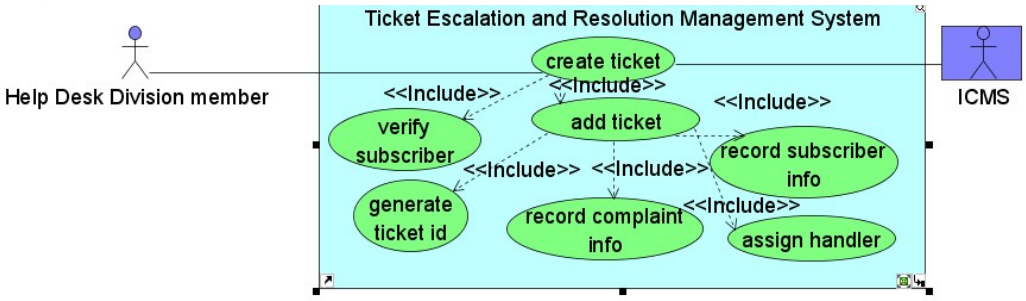


Figure 17. Create Ticket Use Case

Use Case: Create Ticket

Primary actor(and initiator) : Help Desk Division employee

Supporting actor : Integrated Customer Management System (ICMS)

Precondition: The Help Desk Employee is available.

Primary scenario:

1. Help Desk Division employee inputs the subscriber account into the system.

2. The system contacts the ICMS to verify if the account is a valid subscriber account of PLDT's DSL services.
3. Help Desk Division employee inputs the complaint information.
4. TERMS retrieves subscriber information from ICMS.
5. TERMS assigns handler of the ticket as the Help Desk Division employee.
6. The Help Desk Division employee handles the complaint. That ends the scenario P0.

Postcondition:

P0: The Help Desk Division employee handles the complaint; the complaint is recorded.

P1: The Help Desk Division employee is available; the complaint is discarded.

Variants:

2a: The account is not a valid DSL account; this ends the session P1.

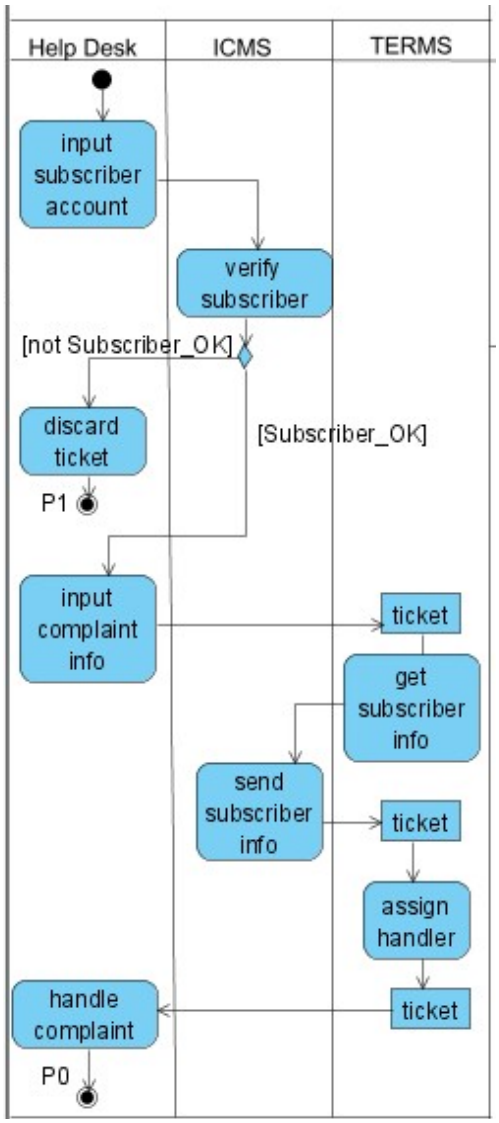


Figure 18. Activity Diagram for Create Ticket Use Case

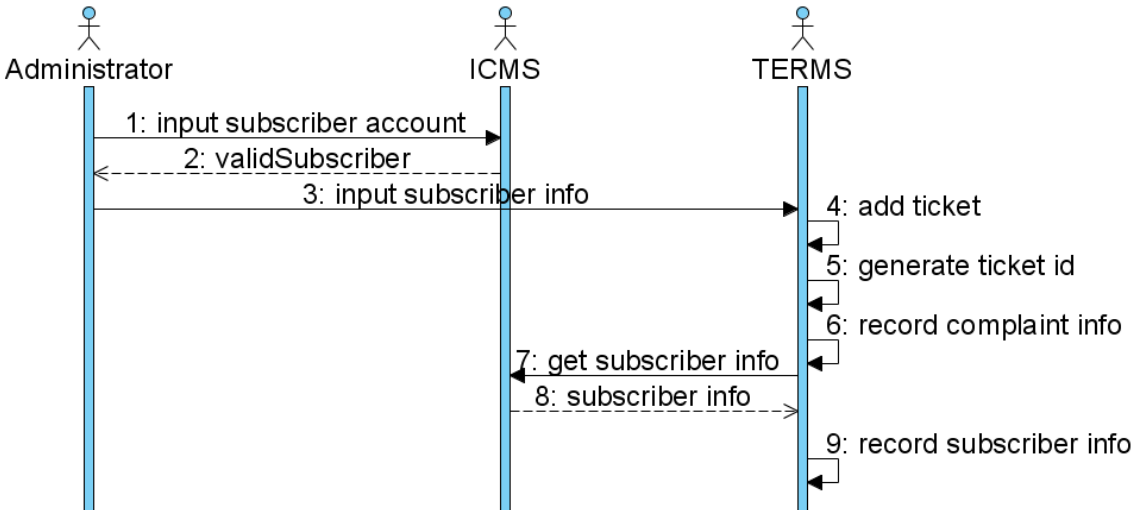


Figure 19. Sequence Diagram for Create Ticket Use Case

2. View Ticket

View Ticket Use Case involves any of the following: Help Desk Division employee, Equipment Specialist, NETRES Division employee and NMOSS Division employee. Indication of search criteria is required for viewing complaints. The system then displays the complaint records satisfying the criteria. Figure 20 shows the View Ticket Use Case.

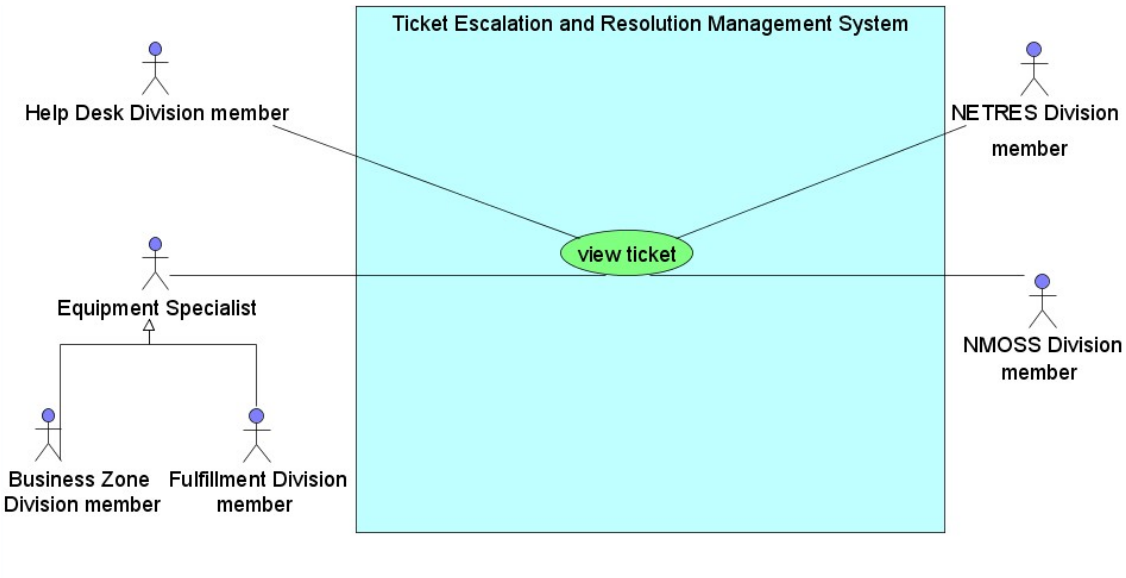


Figure 20. View Ticket Use Case

Use Case: View Ticket

Primary actor(and initiator) : Help Desk Division employee/ Equipment Specialist/ NETRES Division employee/ NMOSS Division employee

Supporting actor : none

Precondition: Help Desk Employee/ Equipment Specialist/ NETRES/ NMOSS Division employee is available.

Primary scenario:

1. Help Desk Employee/ Equipment Specialist/ NETRES/ NMOSS Division employee indicates search criteria.
2. Tickets satisfying the criteria are displayed. That ends the scenario P0.

Postcondition:

P0: Help Desk Employee/ Equipment Specialist/ NETRES/ NMOSS Division employee is available.

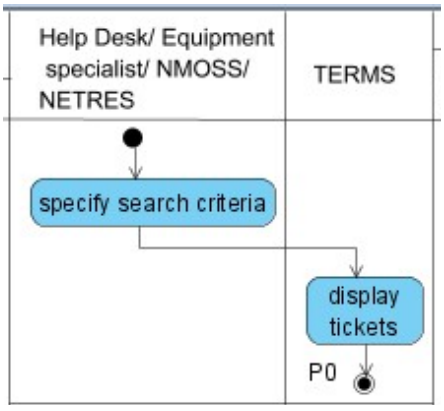


Figure 21. Activity Diagram for View Ticket Use Case

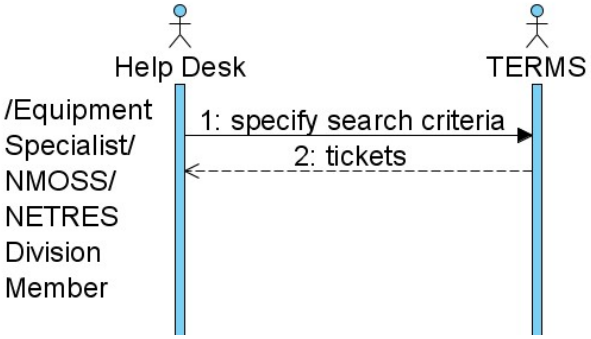


Figure 22. Sequence Diagram for View Complaint Use Case

3. Close Ticket

Only Help Desk Division members or Equipment specialists handling a particular ticket can close it. Tickets are closed only when the problem has been solved by the Help Desk Division member or Equipment specialist handling the ticket. Figure 23 shows the Close Ticket Use Case.

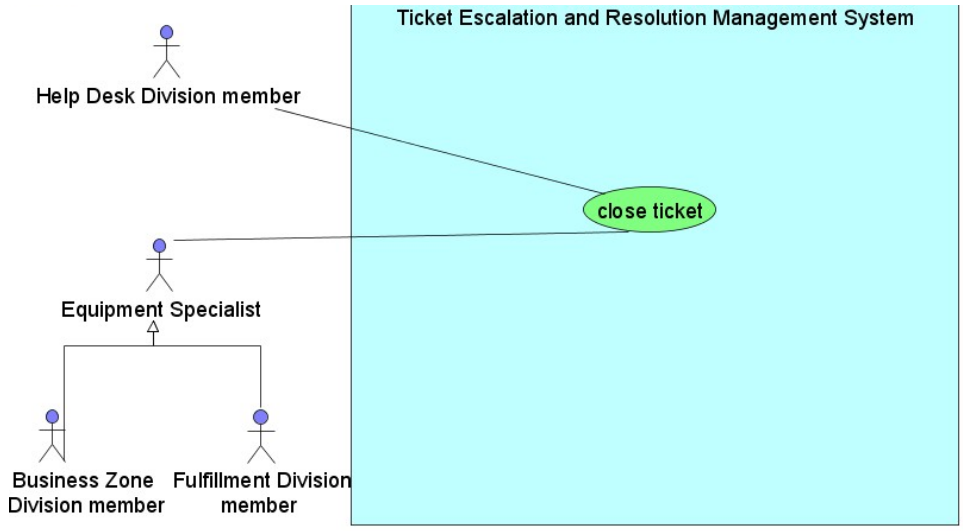


Figure 23. Close Ticket Use Case

Use Case: Close Ticket

Primary actor(and initiator) : Help Desk Division employee/ Equipment Specialist

Supporting actor : none

Precondition : Help Desk/ Equipment Specialist handles the ticket. The Help Desk/Equipment Specialist solves the complaint.

Primary scenario :

1. Help Desk/Equipment Specialist closes the ticket.
2. The ticket is closed. That ends the scenario P0.

Postcondition:

P0: The Help Desk/Equipment Specialist is available. The problem is solved.

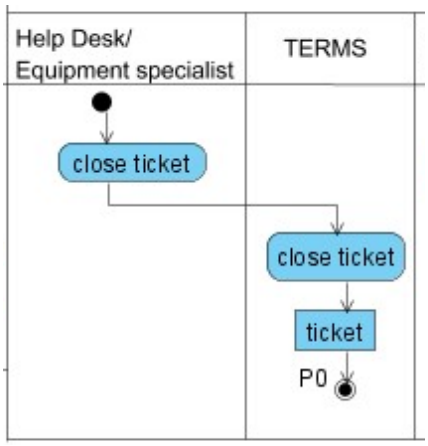


Figure 24. Activity Diagram for Close Ticket Use Case

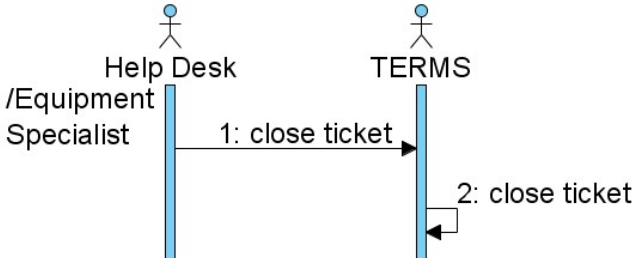


Figure 25. Sequence Diagram Close Ticket Use Case

4. Endorse Ticket

A Help Desk Division Member or Equipment Specialist handling a particular ticket can endorse another member from his/her respective division to handle all tickets assigned to him/her. He/she can only endorse another member if he/she cannot attend to his/her tickets. The endorsed member becomes the new handler of the tickets. Figure 26 shows the Endorse Ticket Use Case.

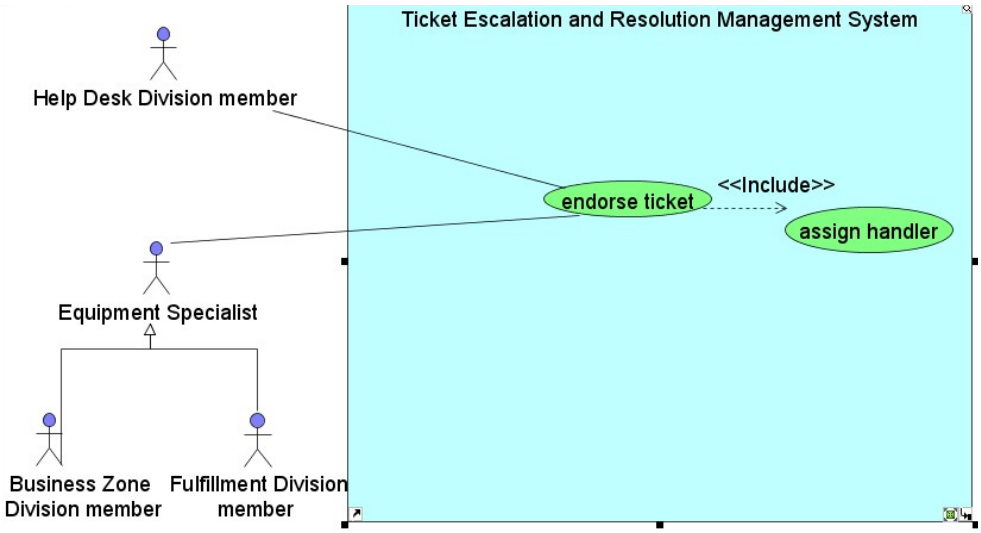


Figure 26. Endorse Ticket Use Case

Use Case: Endorse Ticket

Primary actor(and initiator) : Help Desk/ Equipment specialist

Supporting actor :

Precondition: Help Desk/ Equipment Specialist handles the ticket. Help Desk/ Equipment specialist is unable to attend to his/her tickets.

Primary scenario:

1. Help Desk/ Equipment specialist endorses ticket to another Help Desk/Equipment specialist, respectively.
2. The system assigns the endorsed member as the handler of the ticket. That ends the scenario P0.

Postcondition:

P0: Help Desk/ Equipment specialist does not handle the ticket. Endorsed member handles the ticket.

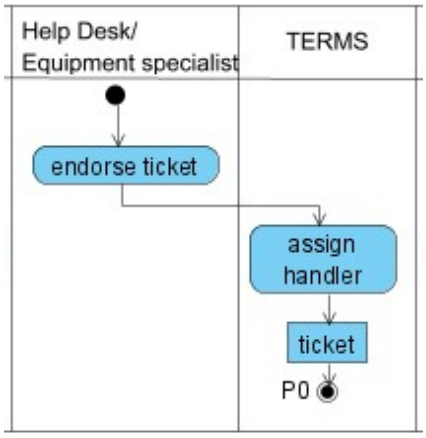


Figure 27 Activity Diagram for Endorse Ticket Use Case

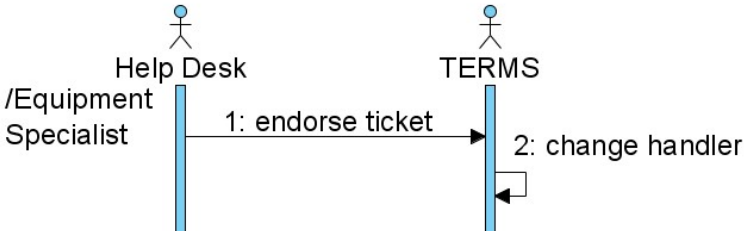


Figure 28. Sequence Diagram for Endorse Ticket Use Case

5. Stage Ticket

The Help Desk Division member or Equipment Specialist handling a particular ticket can stage the ticket. He/she stages a ticket if he/she cannot solve the problem using his/her checking methods. He/she contacts the next more specialized division. Anyone can answer and accept the ticket. The employee from the more specialized who accepted the ticket then becomes the next handler of the ticket after the employee from the lower division performs this process. Figure 29 shows the Stage Ticket Use Case.

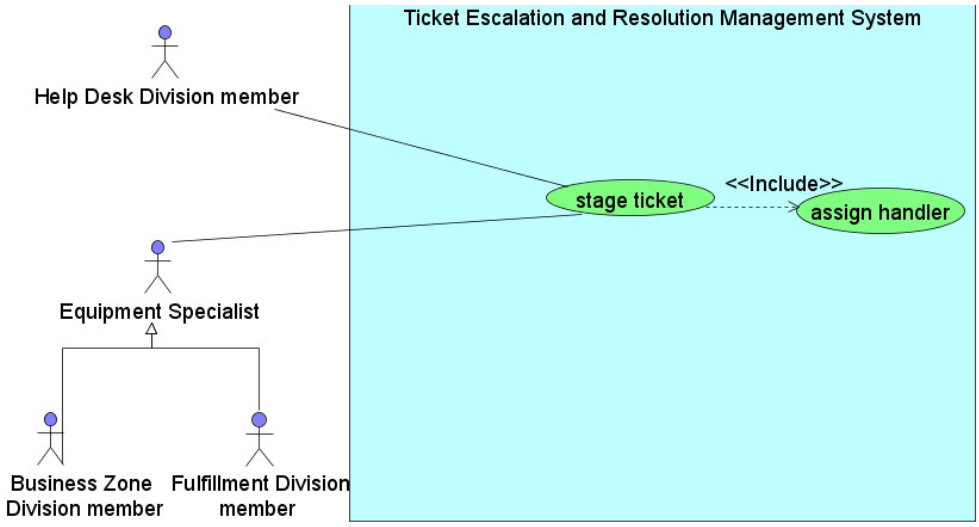


Figure 29. Stage Ticket Use Case

Use Case: Stage Ticket

Primary actor(and initiator) : Help Desk/ Equipment specialist

Supporting actor : none

Precondition : Help Desk/ Equipment specialist handles the ticket. Help Desk/ Equipment specialist cannot solve the problem.

Primary scenario:

1. Help Desk/ Equipment specialist stages ticket to the employee.
2. The system assigns the employee as handler of the ticket. That ends the scenario P0.

Postcondition:

P0: Help Desk/ Equipment specialist does not handle the ticket. Contacted employee from the higher division handles the ticket.

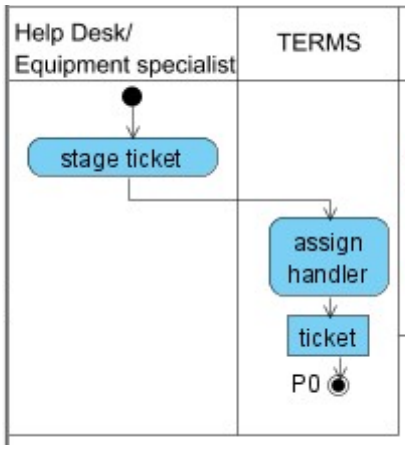


Figure 30. Activity Diagram for Stage Ticket Use Case

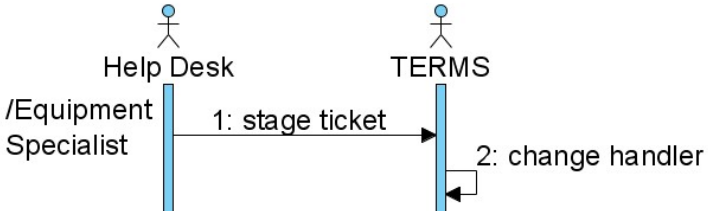


Figure 31. Sequence Diagram for Stage Ticket Use Case

6. Edit Ticket

A NMOSS Division member can edit a particular closed ticket. He/she reads the remarks from the final handler of the ticket. He/she then defines the standardized source of the problem and its resolution and updates the database. Figure 32 shows Edit Complaint Use ticket.

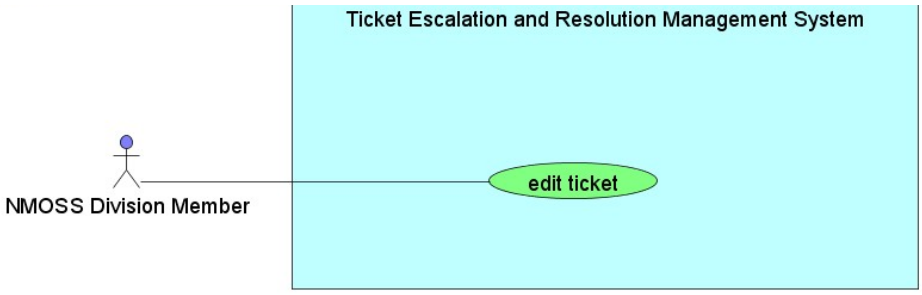


Figure 32. Edit Ticket Use Case

Use Case: Edit Ticket

Primary actor(and initiator) : NMOSS Division member

Supporting actor : none

Precondition: NMOSS Division member is available. Ticket is closed.

Primary scenario:

1. The NMOSS Division member gets the final remarks of the complaint.
2. The NMOSS Division member defines the origin of the problem and responsible group to solve the problem(dis) and its cause and solution(cas). That ends the scenario P0.

Postcondition:

P0: NMOSS Division member is available. Ticket has standardized source of problem and resolution.

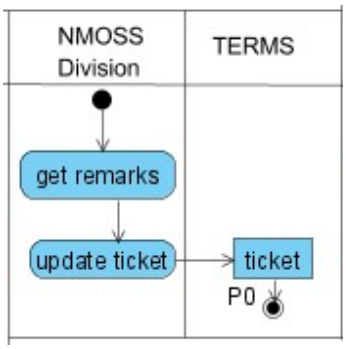


Figure 33. Activity Diagram for Edit Ticket Use Case

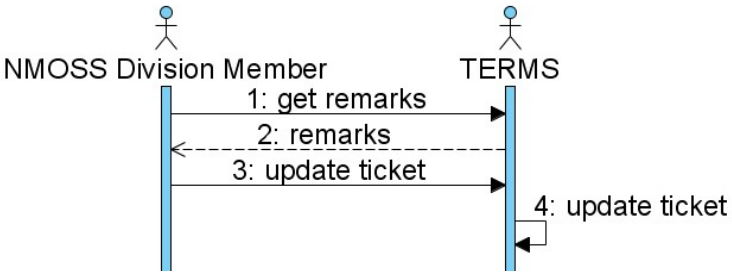


Figure 34. Sequence Diagram for Edit Ticket Use Case

7. Generate Reports

NMOSS or NETRES Division employees can access report generation features of TERMS. The data source for the generated report only includes closed DSL tickets. The generated report is in the form of a squarified treemap summarizing the distribution of complaints. The NMOSS or NETRES Division employee can define settings for the treemap report. Figure 35 shows the Generate Reports Use Case.

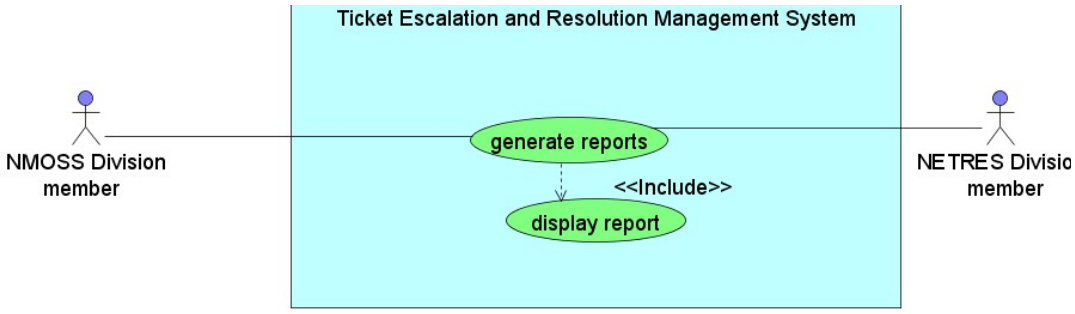


Figure 35. Generate Reports Use Case

Use Case: Generate Reports

Primary actor(and initiator) : NMOSS/ NETRES Division employee

Supporting actor : none

Precondition: NMOSS/NETRES Division employee is available. NMOSS/NETRES Division employee defines desired settings for viewing the report.

Primary scenario:

1. NMOSS/NETRES Division employee views the report and uses its features. That ends the scenario P0.

Postcondition:

P0: NMOSS/NETRES Division employee is available. NMOSS/NETRES Division employee can define new set of settings for viewing the report.

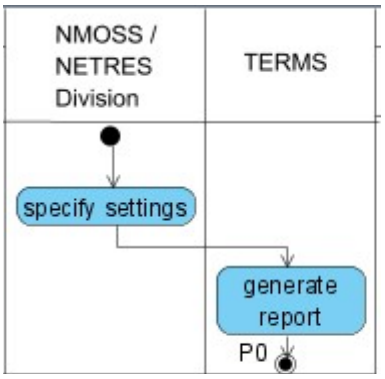


Figure 36. Activity Diagram for Generate Reports Use Case

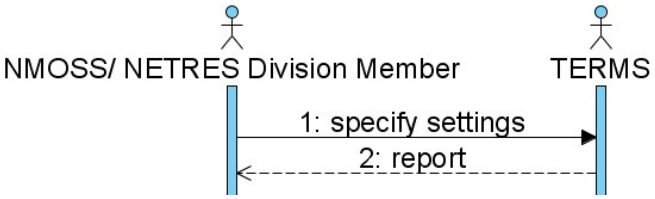


Figure 37. Sequence Diagram for Generate Reports Use Case

8. Manage Users

Administrator can add any type of user and edit user information from TERMS. Figure 38 shows the Manage Users Use Case.

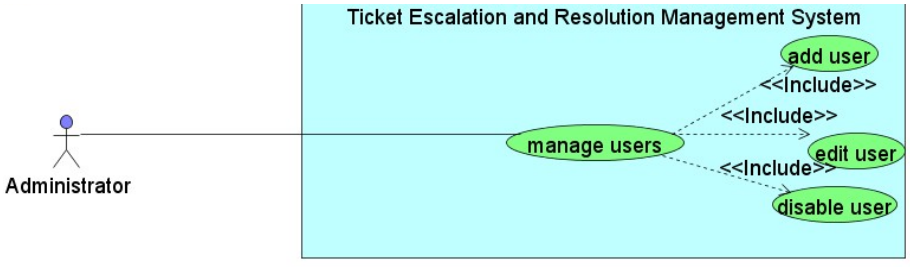


Figure 38. Use Case Diagram for Manage Users Use Case

Use Case: Manage Users

Primary actor(and initiator) : Administrator

Supporting actor : none

Precondition: Administrator is available. User to be disabled does not handle any ticket.

Primary scenario:

1. Administrator defines changes to user information.

Postcondition:

P0: Administrator is available. User information is updated.

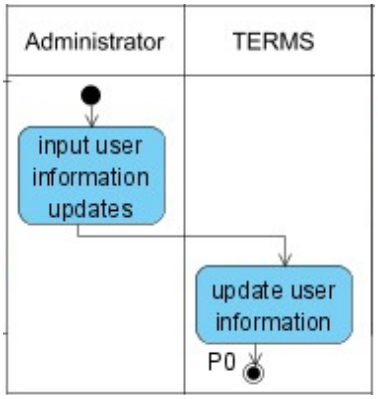


Figure 39. Activity Diagram for Manage Users Use Case

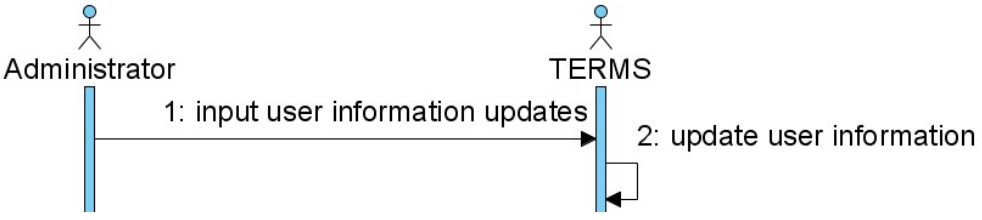


Figure 40. Sequence Diagram for Manage Users Use Case

9. Enable/disable HTTPS for login

Administrators can enable or disable HTTPS for login. Figure 41 shows the Enable/disable HTTPS for login Use Case.

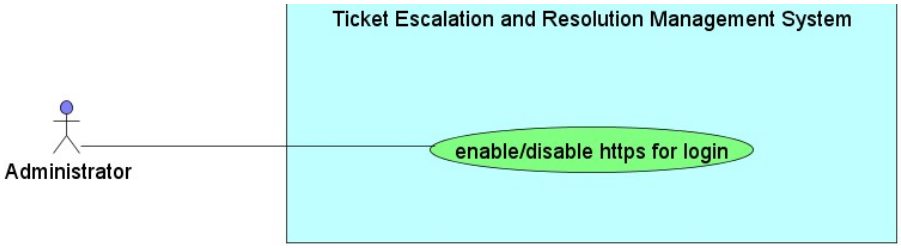


Figure 41. Enable/disable HTTPS for login Use Case

Use Case: Enable/disable HTTPS for login

Primary actor(and initiator) : Administrator

Supporting actor : none

Precondition: Administrator is available.

Primary scenario:

1. Administrator enables/disables HTTPS for login. That ends the scenario P0.

Postcondition:

P0: Administrator is available. HTTPS settings for login are changed.

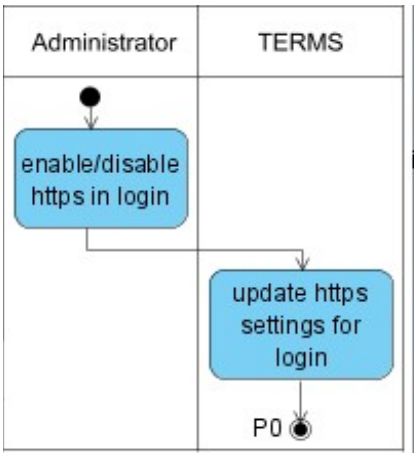


Figure 42. Activity Diagram for Enable/disable HTTPS for login Use Case

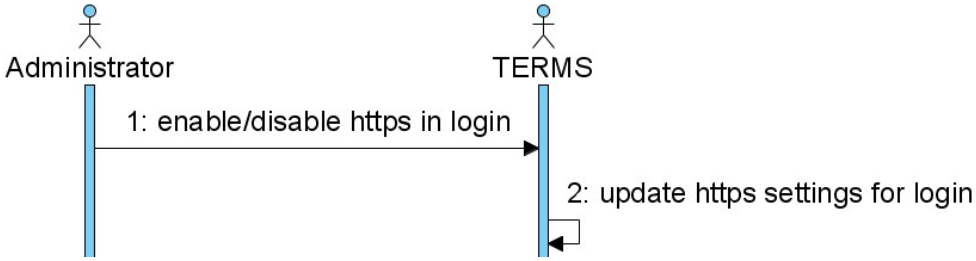


Figure 43. Sequence Diagram for Enable/disable HTTPS for login Use Case

C. Entity Relationship Diagram

There are mainly two tables in the database of TERMS: ticket table and tracking table. The ticket table stores the main information about a particular complaint as well as subscriber information. The tracking table stores the users of the system who handled the ticket. Another table is allocated for storing user information. Figure 44 shows the Entity Relationship Diagram for TERMS.

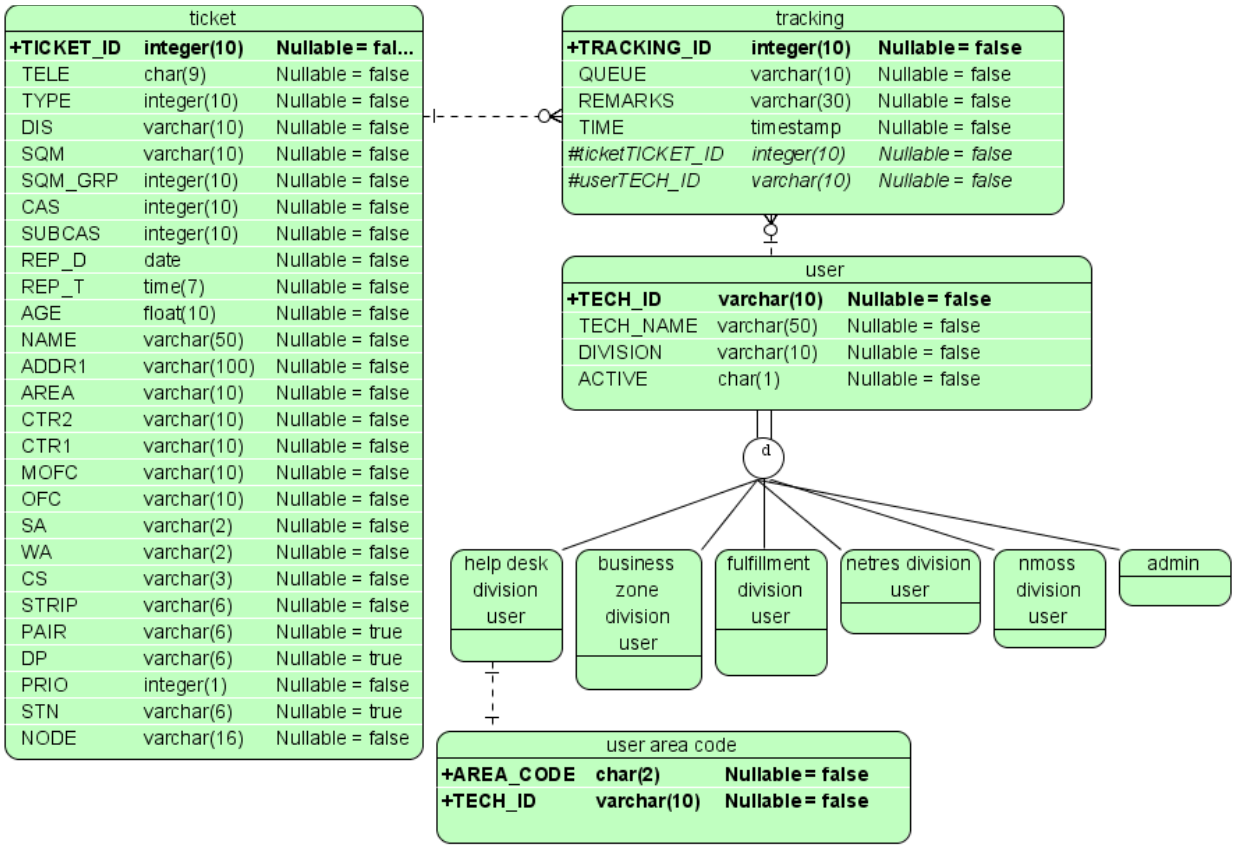


Figure 44. Entity Relationship Diagram for TERMS

D. Data Dictionary

The ticket table has the following attributes:

Attribute	Data Type	Description
TICKET_ID	Integer(10)	
TELE	Char(9)	telephone number of the complainant
TYPE	Integer(10)	id of the nature of complaint/trouble
DIS	Varchar(10)	id of the origin of the problem and responsible group to solve the problem
SQM	Varchar(10)	id of the specific network element/segment involved
SQM_GRP	Integer(10)	id of the general network element/segment involved
CAS	Integer(10)	id of the general cause and solution to the problem
SUBCAS	Integer(10)	id of the specific cause and solution to the problem

REP_D	Date	reported date
REP_T	Time	reported time
AGE	Float(10)	duration in hours when the ticket was handled
NAME	Varchar(50)	subscriber's name
ADDR1	Varchar(100)	subscriber's address
AREA	Varchar(10)	the area covering the subscriber's location
CTR2	Varchar(10)	the region covering the subscriber's location
CTR1	Varchar(10)	the sub-region covering the subscriber's location
MOFC	Varchar(10)	the main office covering the subscriber's location
OFC	Varchar(10)	the office covering the subscriber's location
SA	Varchar(2)	
WA	Varchar(2)	
CS	Varchar(3)	
STRIP	Varchar(6)	
PAIR	Varchar(6)	
DP	Varchar(6)	
PRIO	Integer(1)	
STN	Varchar(6)	
NODE	Varchar(16)	the network node/access gateway id

The tracking table has the following attributes:

Attribute	Data Type	Description
TRACKING_ID	Integer(10)	
TICKET_ID	Integer(10)	
TECH_ID	Varchar(10)	personnel handling the ticket
TIME	Timestamp	date and time the employee started handling the ticket
QUEUE	Integer(10)	This has 2 meanings based on the level of the tracking record. - division the handler belongs to (if the tracking record is at the Help Desk, Business Zone or Fulfillment Division level) - type of network problem (if the tracking record is at the NETRES Division level)
REMARKS	Varchar(10)	

The user table has the following attributes:

Attribute	Data Type	Description
TECH_ID	Varchar(10)	
TECH_NAME	Varchar(50)	Employee's name

DIVISION	Varchar(10)	Employee's division
ACTIVE	Char(1)	Indicator for disabled/enabled users

The user_area_code table has the following attributes:

Attribute	Data Type	Description
TECH_ID	Varchar(10)	
AREA_CODE	Char(2)	Area code assigned to employee

E. Treemap Report Generation Algorithm

In generating the treemap report, the following are the inputs of NMOSS and NETRES:

1. hierarchy to display and order
2. attribute of basis for the size of the nodes
3. attribute of basis for the color of the nodes
4. width of the padding between the subdivisions
5. size of the font for labeling
6. filter criteria

NETRES Division has the option of having an additional input of span of time to cover

TERMS Treemap Generation Algorithm:

Definition of terms:

attribute – property of each complaint group

attribute classification – specific type of attribute

attribute weight – numerical value of a particular attribute classification

generate_treemap – function which is recursively called for each treemap to be drawn

hierarchy – sequence of attributes input by the user

attribute – a data type containing the attribute name

area – a data type which has the following fields : length, width, x, y

map – a data type which has the following fields : attribute name, attribute classification, attribute classification weight

areaMap – a data type which has the following fields : area, map

squarify function – procedure lays out the rectangles in horizontal and vertical rows

width() – gives the length of the shortest side of the remaining subrectangle in which the current row is placed

layoutrow() – adds a new row of children to the subrectangle

++ – used to denote concatenation of lists

[x] – is the list containing element x

[] – is the empty list

worst – gives the highest aspect ratio of a list of rectangles, given the length of the side along which they are to be laid out

The following are all the attributes and their corresponding attribute weights:

1. the type of node – number of complaints for each type of node
2. the nature of complaint – the number of complaints for each nature of complaint

3. the general disposition – the number of complaints for each general disposition
4. the specific disposition – the number of complaints for each specific disposition
5. the general network element/segment involved – the number of complaints for each general network element/segment involved
6. the specific network element/segment involved – the number of complaints for each specific network element/segment involved
7. the general cause and solution – the number of complaints for each general cause and solution
8. the specific cause and solution – the number of complaints for each specific cause and solution
9. the area – the number of complaints for each area
10. the average length of time of service – the average length of time of service for the complaint group

```
function generate_treemap(attribute[] hierarchy, area boundingArea, queryString query)
{
	currentAttribute = head(hierarchy);
	map weights[] = getWeights(currentAttribute, query);
	sort(weights, descending);
	scale(weights, boundingArea);
	squarify(hierarchy, query, weights, [], min(boundingArea.width,
boundingArea.length), boundingArea);
}

function squarify(attribute[] hierarchy, queryString query, map[] weights, areaMap[]
row, real width, area boundingArea){
	map currentWeight = head(weights);
```

```
        if worst(row,width) ≤ worst(row++[currentWeight],width){
                update(row, currentWeight);
                squarify(hierarchy, query, tail(weights), row, width, boundingArea);
        }else{
                layoutrow(hierarchy, query, row);
                squarify(hierarchy,    query,    weights,    [],    boundingArea.width(),
boundingArea);
        }
}

function layoutrow(attribute[] hierarchy, queryString query, areaMap[] row){
        for each ith element of row{
                plot(row[i]);
                generate_treemap(tail(hierarchy), row[i].area, query+row[i].map);
        }
}
```

F. System Architecture

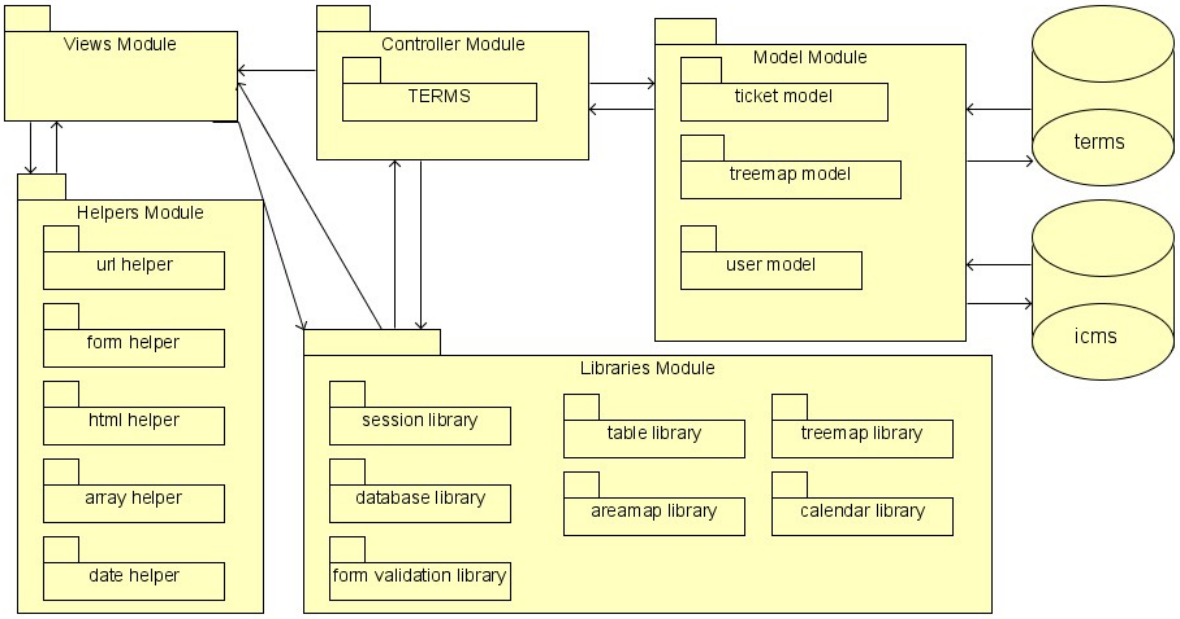


Figure 45. System Architecture Diagram for TERMS

TERMS Modules:

1. View Module – this module is contains information to be displayed in web pages.

2. Controller Module – this module is responsible for coordinating the View Module and the Model Module.

3. Model Module – this module is responsible for retrieving and updating information in the database.

Model Modules:

1. Ticket Model – this model module is responsible for ticket operations on the database such as adding tickets, editing and retrieving ticket information.

2. Treemap Model – this model module is responsible for generating the treemap report of the system using the Squarified Treemap algorithm. The Treemap Model retrieves ticket data from the database for generating the reports. It handles displaying treemaps on the screen and on files.

3. User Model – this model module is responsible for checking user validity and creating, updating, disabling and viewing user accounts.

The TERMS uses the CodeIgniter framework, a PHP framework based on the Model-View-Controller(MVC) development pattern. This framework divides the code into three groups, namely the Models, Views and Controllers. Database operations such as fetching, updating and adding records are performed at the Model classes. The Views contains information shown to users. The Controller coordinates the Models, Views and other resources needed to generate web pages. [17]

G. Technical Requirements

Server

- Ubuntu Linux V 8.04
- Apache Web Server V 2.2.8
- PHP 5.3
- Imagick V3.0 (Pecl Module)
- ImageMagick V6.5.9-8
- Spreadsheet_Excel_Writer Package v. 0.9.2 (Pear Module)
- MySQL 5.0.51a

Client

- Web browser (Firefox V3.5.2, Chrome V 4.0, Safari V 4.0.4, Internet Explorer V8)
- Adobe SVG Viewer plugin for SVG (for IE)

V. Results

The Login page allows users of the system to log into the system. Figure 46 shows the login page.

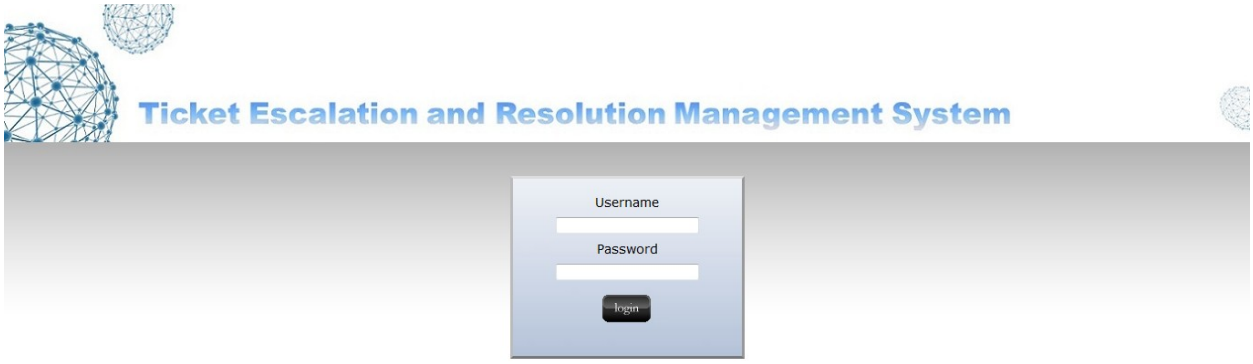


Figure 46. Login Page

The Add Ticket page allows the Help Desk Division employee to add a new ticket into the system. This feature can be accessed by clicking on the Add Ticket tab. Figure 47 shows the form for adding a ticket.

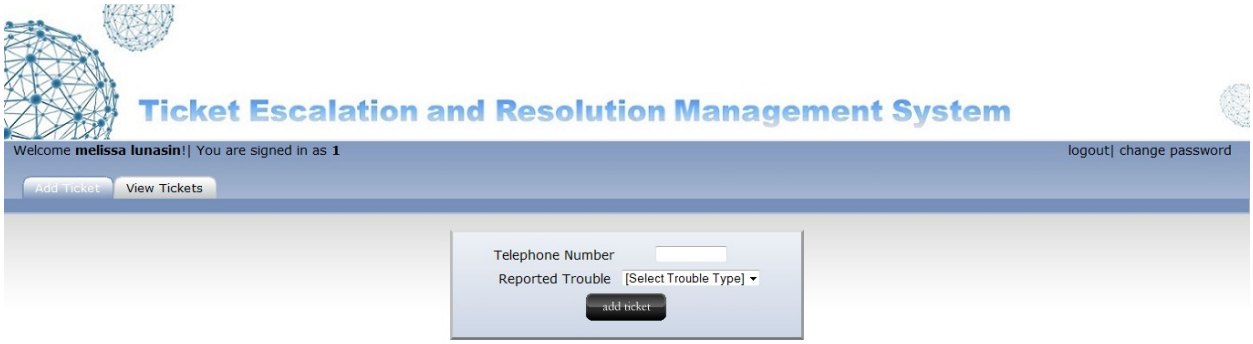


Figure 47. Add Ticket Page

The View Tickets page has three subcategories, Browse My Tickets, Search Tickets and Browse Tickets. For all categories, basic information about each ticket is displayed as well as an option to view each ticket's details.

The Browse My Tickets category allows the Help Desk, Business Zone, Fulfillment or NETRES Division employee to view all tickets assigned to him/her. Figure 48 shows the Browse My Tickets subcategory of the View Tickets page.

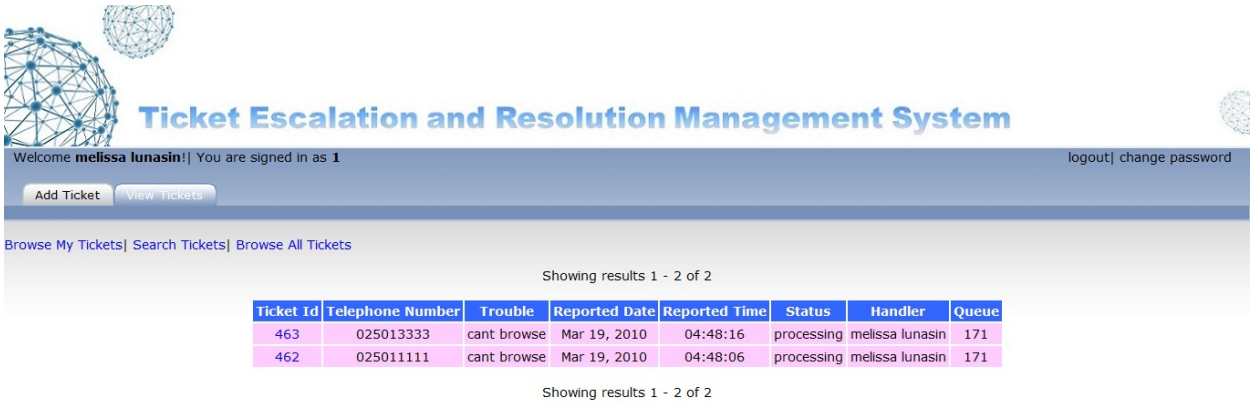


Figure 48 View Tickets Page - Browse My Tickets Category

The Search Tickets category allows all users to search through all tickets input into the system by specifying the criteria. Figure 49 shows the Search Tickets subcategory of the View Tickets page.

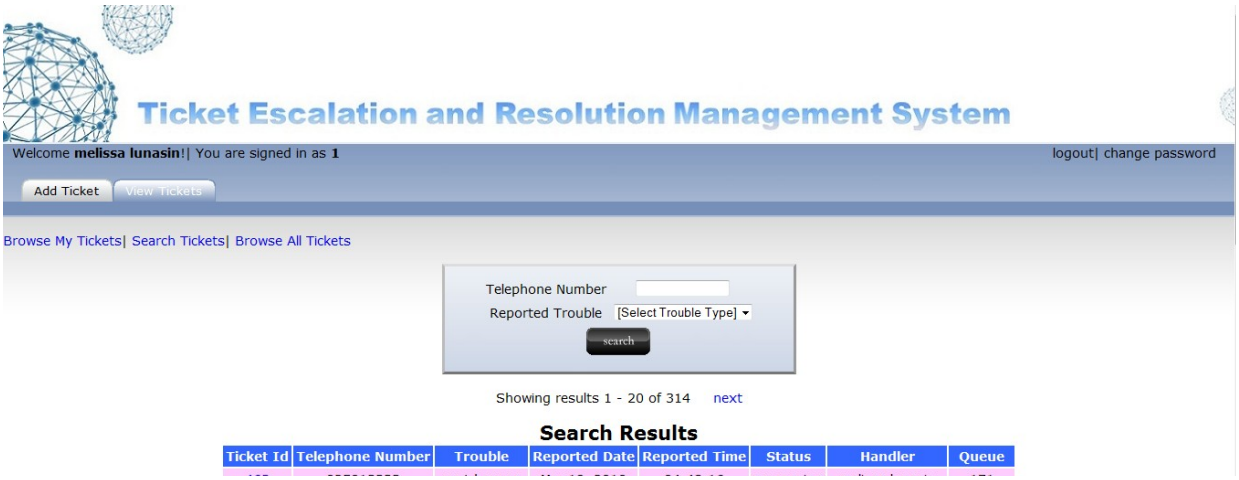
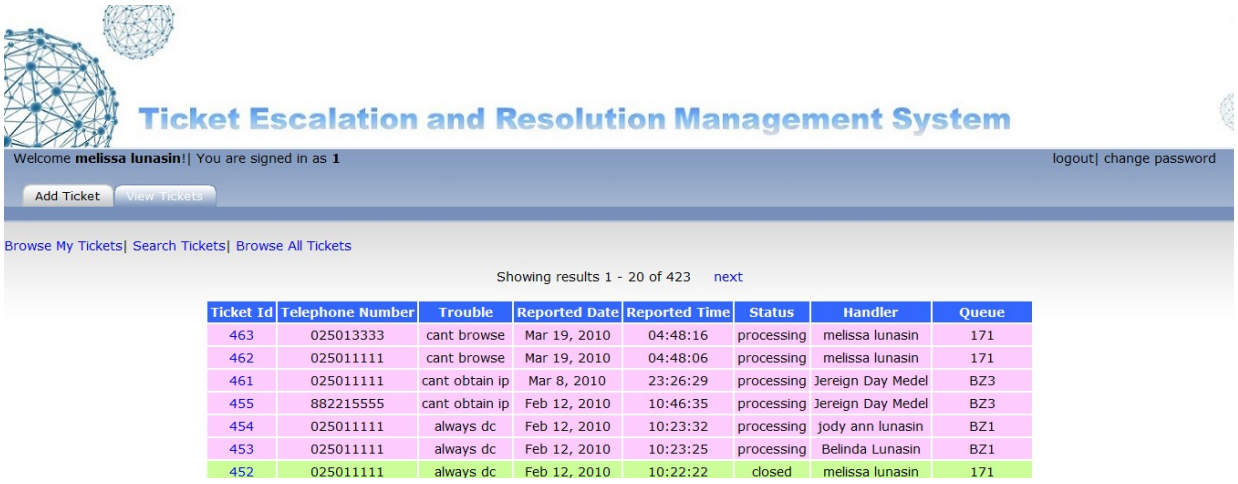


Figure 49. View Tickets Page - Search Tickets Category

Browse Tickets category allows all users to browse through all tickets input into the system. Figure 50 shows the Browse All Tickets subcategory of the View Tickets page.



Ticket Id	Telephone Number	Trouble	Reported Date	Reported Time	Status	Handler	Queue
463	025013333	cant browse	Mar 19, 2010	04:48:16	processing	melissa lunasin	171
462	025011111	cant browse	Mar 19, 2010	04:48:06	processing	melissa lunasin	171
461	025011111	cant obtain ip	Mar 8, 2010	23:26:29	processing	Jereign Day Medel	BZ3
455	882215555	cant obtain ip	Feb 12, 2010	10:46:35	processing	Jereign Day Medel	BZ3
454	025011111	always dc	Feb 12, 2010	10:23:32	processing	jody ann lunasin	BZ1
453	025011111	always dc	Feb 12, 2010	10:23:25	processing	Belinda Lunasin	BZ1
452	025011111	always dc	Feb 12, 2010	10:22:22	closed	melissa lunasin	171

Figure 50 View Tickets Page - Browse Tickets Category

The View Ticket Details Page displays complete information about the selected ticket as well as the ticket history. Options such as stage ticket, endorse ticket, close ticket and edit ticket are available on this page. Figure 51 shows the View Ticket Details Page.

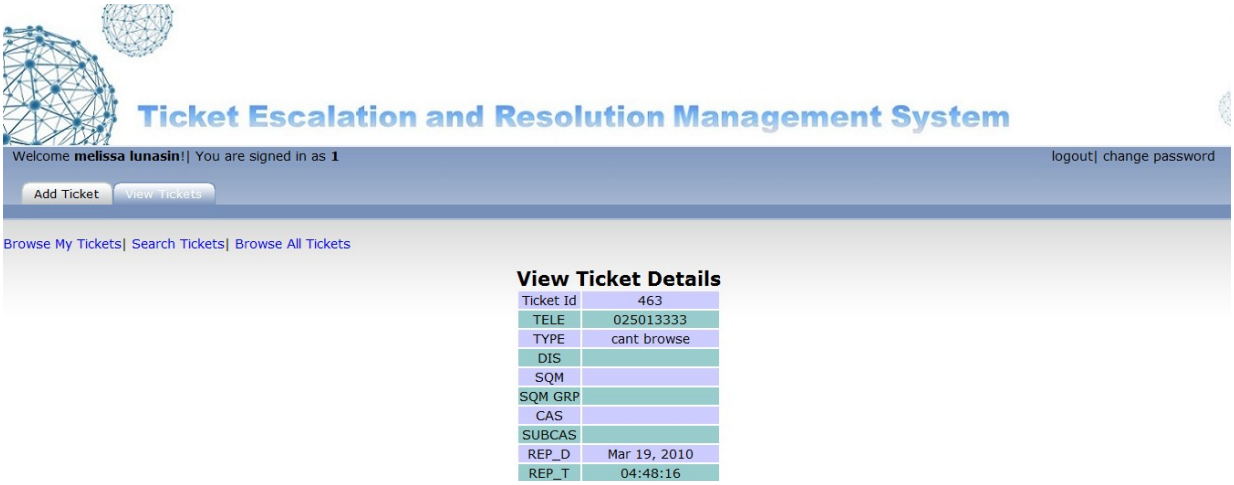


Figure 51 View Ticket Details Page

The Stage Ticket page allows the ticket handler to stage tickets assigned to him/her. Figure 52 shows the Stage Ticket page.

Figure 52. Stage Ticket Page

The Endorse Ticket page allows the ticket handler to endorse tickets assigned to him/her in case he/she could not attend to his/her duties. Figure 53 shows the Endorse Ticket page.

Figure 53. Endorse Ticket Page

The Close Ticket page allows the ticket handler to close tickets assigned to him/her. Figure 54 shows the Close Ticket page.

Figure 54. Close Ticket Page

The Edit Ticket page allows the NMOSS Division employee to specify the DIS and SUBCAS fields of closed tickets. Figure 55 shows the Edit Ticket page.

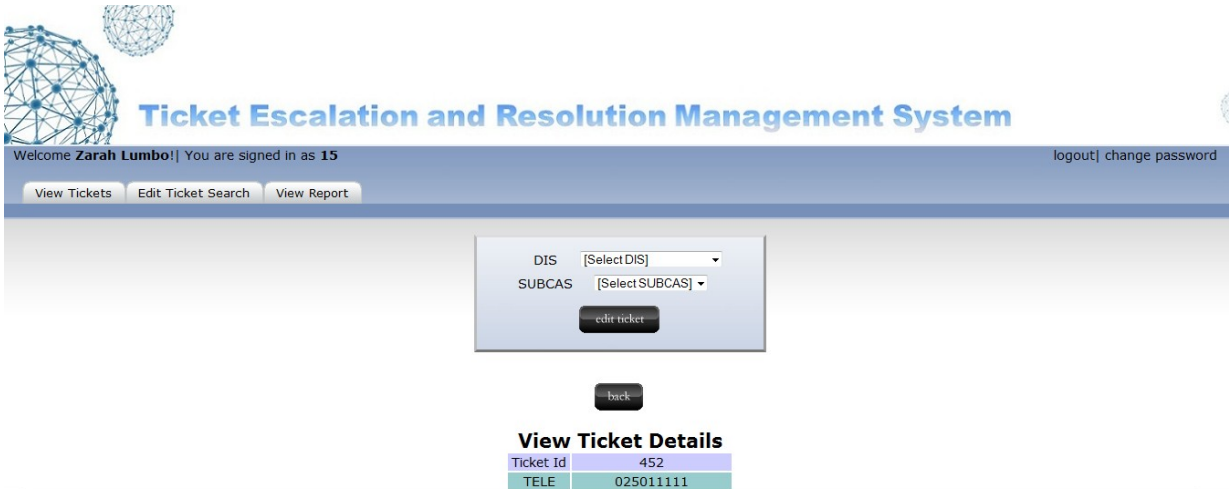


Figure 55. Edit Ticket Page

The View Report page can be accessed by NMOSS and NETRES Division employees to generate treemap reports summarizing the database of complaints. The left portion shows the generated treemap graph and the right part allows viewing of node information as well as specifying of filter criteria and settings. Figure 56 shows the View Report page.

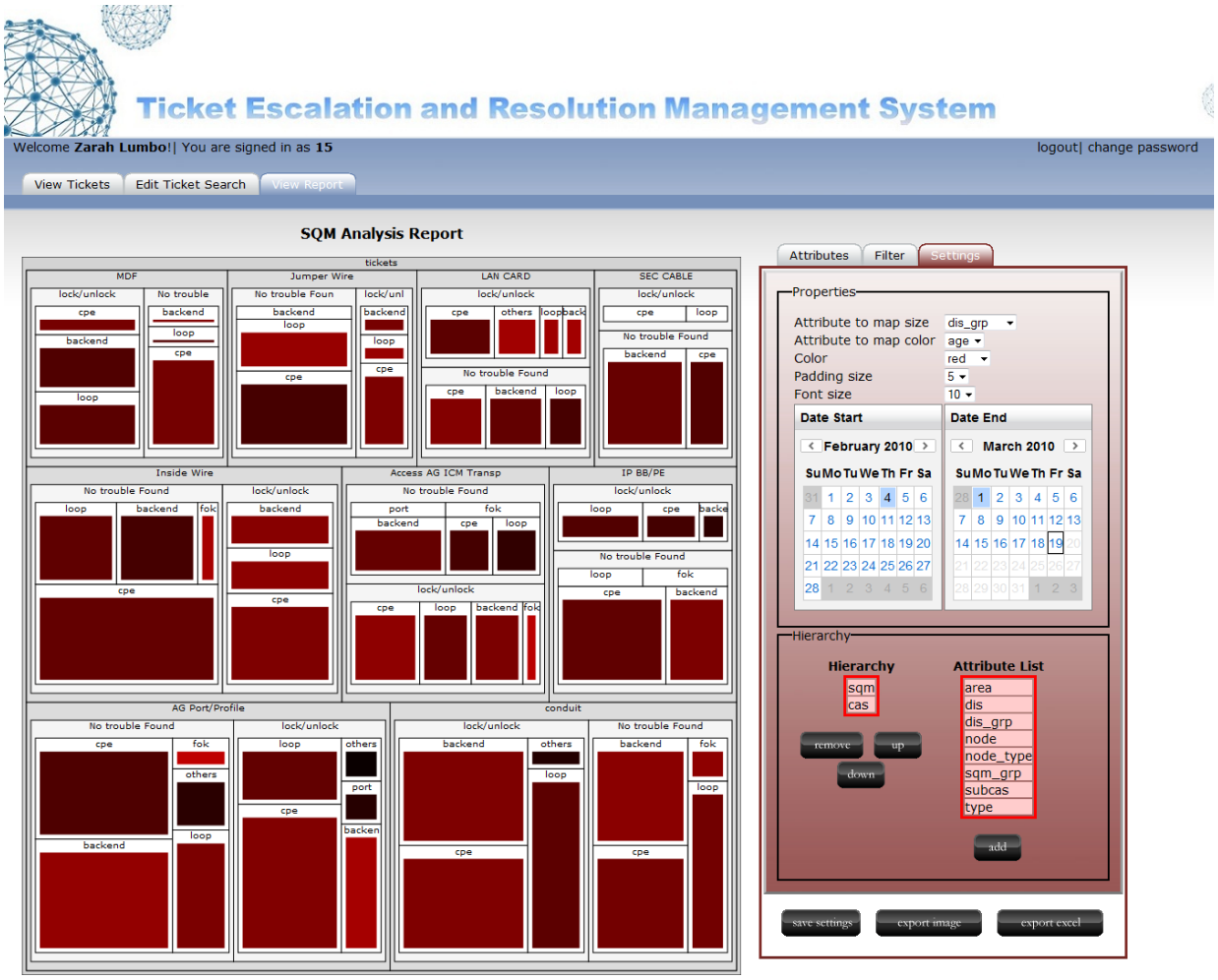


Figure 56. View Report Page

The left portion of the View Report page shows the generated treemap graph. Clicking on any particular region updates the attributes tab and shows information about the clicked region on the Attributes tab of the right portion of the View Report page. Right-clicking on a particular region magnifies it, such that what can be seen are the clicked region, its parent regions, and all regions within it. A middle-click on any region restores the complete treemap graph.

There are three tabs in the right pane of the View Report page, the Attributes tab, Filter tab and Settings tab. Below it are options for saving the defined settings, exporting the generated treemap into a JPEG image and exporting of the selected tickets into a report in excel format.

The Attributes tab shows all attribute values corresponding to the left-clicked region. Figure 57 shows the Attributes tab.

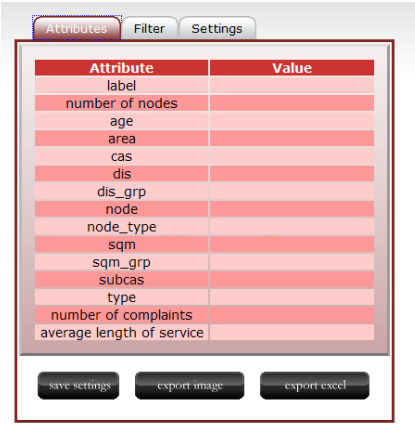


Figure 57. View Report – Attributes Tab

The Filter tab allows the user to select attribute values to be displayed. Figure 58 shows the Filter tab.

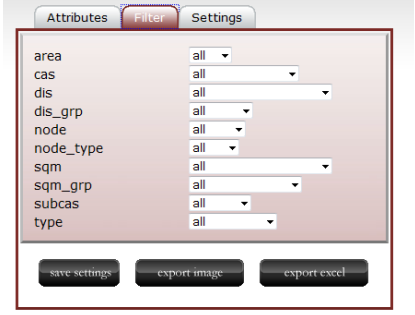


Figure 58.View Report - Filter Tab

The Settings tab allows modification of attribute to map size, attribute to map color, font size, padding size and color. It also allows the user to specify the hierarchy to be displayed. Figure 59 shows the Settings tab.

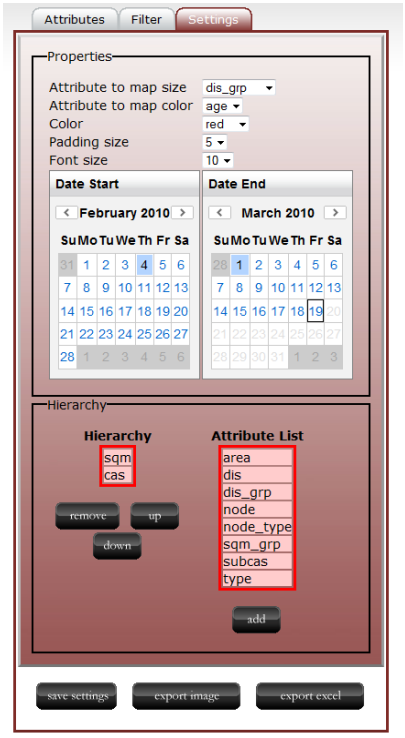


Figure 59. View Report - Settings Tab

The Add User page allows the Administrator to add users into the system. This feature can be accessed from the User Management Tab. Figure 60 shows the Add User page.

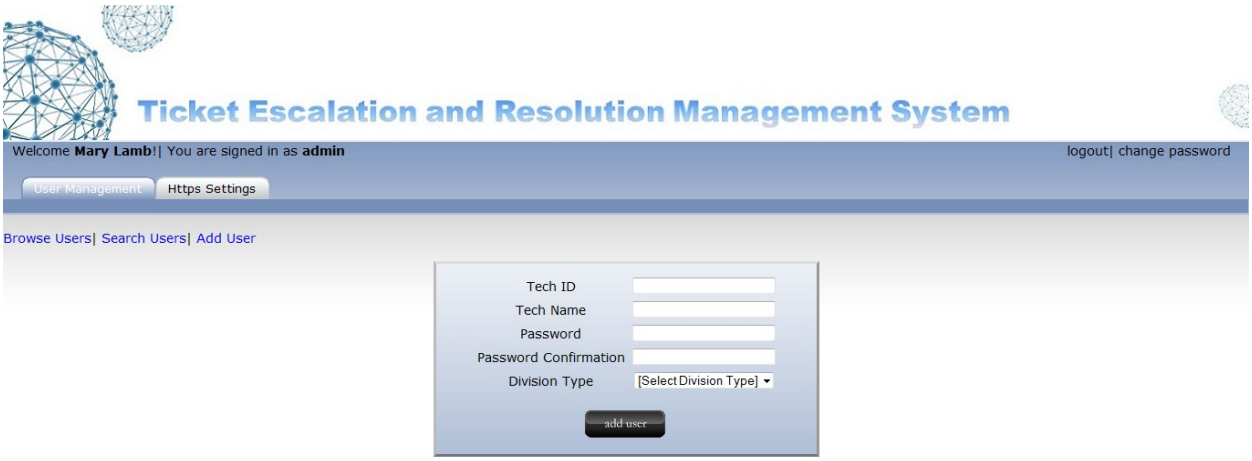


Figure 60. Add User Page

The Browse Users page allows the Administrator to browse through all the users into the system. This feature is available from the User Management Tab. Figure 61 shows the Browse Users page.

Figure 61. Browse Users Page

The Search User page allows the Administrator to search through all the users of the system by specifying the Tech ID or Tech Name field. This can also be accessed from the User Management Tab. Figure 62 shows the Search User Page.

Figure 62. Search User Page

The Edit User page allows the Administrator to edit any information of all users of TERMS except for the Tech ID. This feature can be accessed by clicking on the Tech ID link on the Browse Page or on the User Search results. Figure 63 shows the Edit User page.

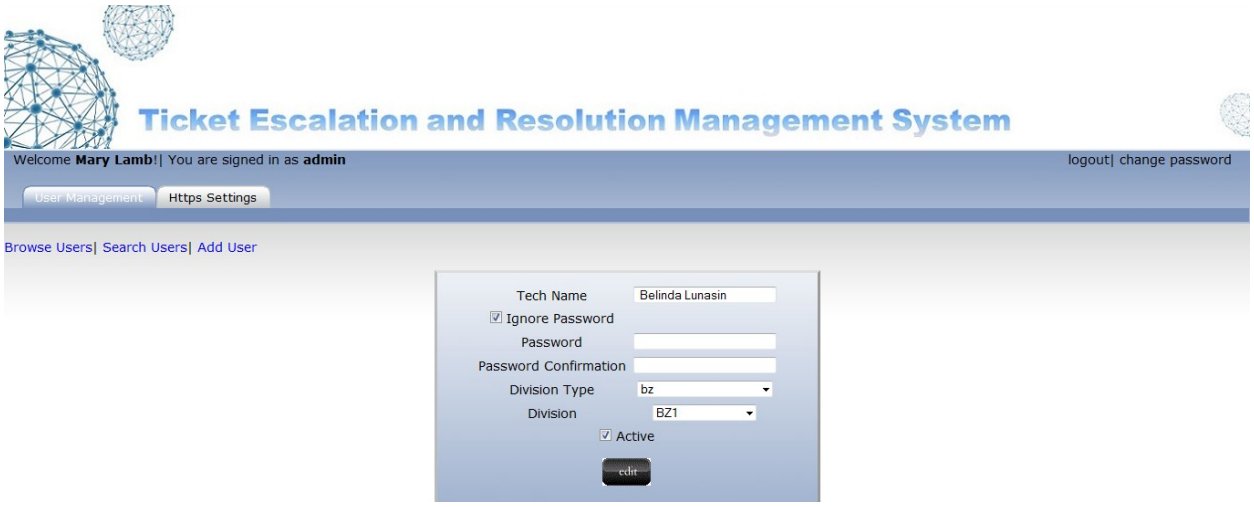


Figure 63. Edit User Page

The HTTPS Settings page allows the Administrator to enable or disable HTTPS settings for login. Figure 64 shows the HTTPS Settings page.

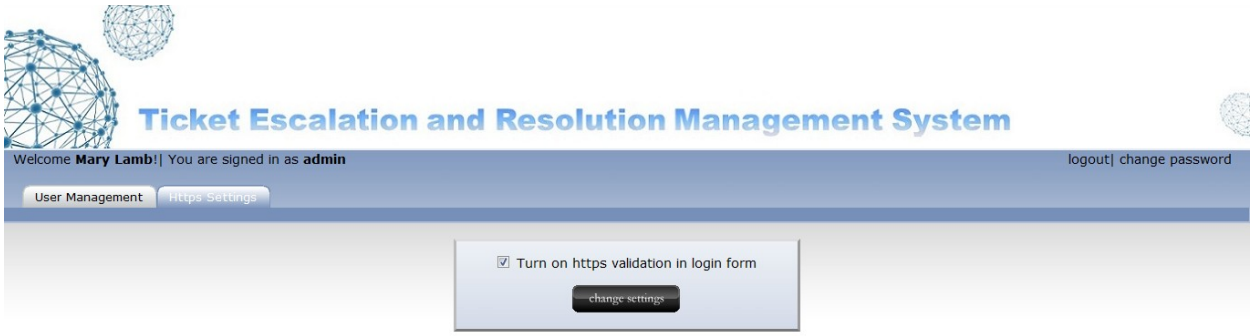


Figure 64. HTTPS Settings Page

The Change Password page allows any user to change his/her password. Figure 65 shows the Change Password page.

Figure 65. Change Password Page

VI. Discussions

TERMS (Ticket Escalation and Resolution Management System) has 4 types of users, the Help Desk Division Employee, Equipment Specialist, NETRES Division Employee, the NMOSS Division Employee. It interacts with the ICMS for customer account verification. The Equipment specialist has two types, the Business Zone Division member and the Fulfillment Division member.

Help Desk Division Members can record complaints, view complaint records, close tickets, endorse ticket and stage tickets. The complaint recording process involves retrieving information from the ICMS. Equipment Specialists can view tickets, close tickets, endorse members and stage tickets. NETRES Division members can view tickets and generate reports. NMOSS Division Members can view tickets, edit tickets and generate reports.

Creating a ticket involves the Help Desk Division Member, ICMS and TERMS. Recording the complaint involves verifying if the caller is a valid subscriber and adding the complaint. The process of verifying the subscriber is done by accessing the ICMS, the system for storing subscriber information. Adding the complaint includes automatically generating a ticket id, recording complaint information, recording subscriber information and assigning the handler. A unique ticket id is assigned to each ticket. Complaint information such as nature of the complaint and date and time when the complaint was filed are stored. Subscriber information which is stored in the ICMS is retrieved and added to the complaint record. The handler is assigned by the system to be the Help Desk Division member who created the ticket.

Viewing a ticket involves any of the following: Help Desk Division employee, Equipment Specialist, NETRES Division employee and NMOSS Division employee. Indication of search criteria is required for viewing complaints. The system then displays the complaint records satisfying the criteria.

Only Help Desk Division members or Equipment specialists handling a particular ticket can close it. Tickets are closed only when the problem has been solved by the Help Desk Division member or Equipment specialist handling the ticket.

A Help Desk Division Member or Equipment Specialist handling a particular ticket can endorse another member from his/her respective division to handle all tickets assigned to him/her. He/she can only endorse another member if he/she cannot attend to his/her tickets. The endorsed member becomes the new handler of the tickets.

The Help Desk Division member or Equipment Specialist handling a particular ticket can stage the ticket. He/she stages a ticket if he/she cannot solve the problem using his/her checking methods. He/she contacts the next more specialized division. Anyone can answer and accept the ticket. The employee from the more specialized who accepted the ticket then becomes the next handler of the ticket after the employee from the lower division performs this process.

A NMOSS Division member can edit a particular closed ticket. He/she reads the remarks from the final handler of the ticket. He/she then defines the standardized source of the problem and its resolution and updates the database.

NMOSS or NETRES Division employees can access report generation features of TERMS. The data source for the generated report only includes closed DSL tickets. The generated report is in the form of a squarified treemap summarizing the distribution of complaints. The NMOSS or NETRES Division employee can define settings for the treemap report.

The Treemap-based report of TERMS is much easier to interpret compared to raw data in tabular form in the excel files. The NETRES or NMOSS Division members can easily spot and compare groups of tickets by simply observing the space they occupy in the treemap. For example, if the rectangle pertaining to tickets with trouble type CAN'T SYNC has the largest area, the NMOSS or NETRES Division employee can deduce that the most frequently reported trouble is CAN'T SYNC. Aside from the attribute to map size, the attribute to map color of the treemap report is also customizable. So aside from getting information from comparing the sizes of each rectangle, the NETRES and NMOSS Division members can also collect information about the color of each rectangle. For example, if the set attribute to map color is the AGE, a darker color means that the average AGE for that particular group of tickets took a much longer time to resolve compared to light colored ones.

The Treemap report includes a hierarchy change feature. This feature allows the NETRES or NMOSS Division member to easily modify and generate treemap reports based on the selected hierarchy. For example, if the selected hierarchy is CAS and then TROUBLE, the tickets are first classified by the CAS field value then further divided into TROUBLE field value. Using this feature, cross-sectional data can be easily be generated and can be useful in observing trends.

Various options are available to customize the treemap report according to the needs of the NETRES or NMOSS Division member. Aside from the attribute to map size and attribute to map color, the report's date settings can also be modified. This can be performed by selecting two dates, and then the generated report covers all the days within the selected dates. Also, the color, font size and padding size can be changed by selecting a value from their corresponding list of choices. Filter criteria can also be specified. For example, choosing the CAN'T BROWSE option in the filter TROUBLE dropdown will generate a treemap wherein all the included tickets have a TROUBLE type value of CAN'T BROWSE.

Left-clicking on a particular rectangle displays all information concerning the group of tickets represented by the selected rectangle. The Treemap report can be zoomed by right-clicking on the desired rectangle. This magnifies the selected rectangle, such that only all the classifications above it and within it will be displayed. Middle-clicking then restores the treemap to its original form.

VII. Conclusion

The Ticket Escalation and Resolution Management System has 4 types of users, the Help Desk Division Employee, Equipment Specialist, NETRES Division Employee, the NMOSS Division Employee. It interacts with the ICMS for customer account verification. The Equipment specialist has two types, the Business Zone Division member and the Fulfillment Division member.

Help Desk Division Members can record complaints, view complaint records, close tickets, endorse ticket and stage tickets. The complaint recording process involves retrieving information from the ICMS. Equipment Specialists can view tickets, close tickets, endorse members and stage tickets. NETRES Division members can view tickets and generate reports. NMOSS Division Members can view tickets, edit tickets and generate reports.

Creating a ticket involves the Help Desk Division Member, ICMS and TERMS. Recording the complaint involves verifying if the caller is a valid subscriber and adding the complaint. The process of verifying the subscriber is done by accessing the ICMS, the system for storing subscriber information. Adding the complaint includes automatically generating a ticket id, recording complaint information, recording subscriber information and assigning the handler. A unique ticket id is assigned to each ticket. Complaint information such as nature of the complaint and date and time when the complaint was filed are stored. Subscriber information which is stored in the ICMS is retrieved and added to the complaint record. The handler is assigned by the system to be the Help Desk Division member who created the ticket.

Viewing a ticket involves any of the following: Help Desk Division employee, Equipment Specialist, NETRES Division employee and NMOSS Division employee. Indication of search criteria is required for viewing complaints. The system then displays the complaint records satisfying the criteria.

Only Help Desk Division members or Equipment specialists handling a particular ticket can close it. Tickets are closed only when the problem has been solved by the Help Desk Division member or Equipment specialist handling the ticket.

A Help Desk Division Member or Equipment Specialist handling a particular ticket can endorse another member from his/her respective division to handle all tickets assigned to him/her. He/she can only endorse another member if he/she cannot attend to his/her tickets. The endorsed member becomes the new handler of the tickets.

The Help Desk Division member or Equipment Specialist handling a particular ticket can stage the ticket. He/she stages a ticket if he/she cannot solve the problem using his/her checking methods. He/she contacts the next more specialized division. Anyone can answer and accept the ticket. The employee from the more specialized who accepted the ticket then becomes the next handler of the ticket after the employee from the lower division performs this process.

A NMOSS Division member can edit a particular closed ticket. He/she reads the remarks from the final handler of the ticket. He/she then defines the standardized source of the problem and its resolution and updates the database.

NMOSS or NETRES Division employees can access report generation features of TERMS. The data source for the generated report only includes closed DSL tickets. The generated report is in the form of a squarified treemap summarizing the distribution of complaints. The NMOSS or NETRES Division employee can define settings for the treemap report.

VIII. Recommendations

TERMS can be improved by integrating a telephone number detection tool to eliminate the process of manually encoding the telephone number upon ticket creation. Automating delegation of tickets wherein the system decides which employee should handle the ticket in consideration and introducing a ticket assignment notification system are possible enhancements to TERMS. A feature which allows users to interact with TERMS via SMS may be added into the system. It would then be more convenient for personnel such as Business Zone Division members whose line of work consists mainly of fieldwork.

IX. Bibliography

[1] *PLDT – About Home*. Retrieved September 8, 2009 from http://www.pldt.com.ph/about/Pages/summary.aspx

[2] Lunasin, Jody Ann. Personal interview. 3 September 2009.

[3] Johnson, B. & Schneiderman, B.. *Tree-Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures.* In Proceedings of the 2nd conference on Visualization '91 (1991): 284-291

[4] Schneiderman, B.. *Tree Visualization with Tree-maps: A 2-d Space-Filling Approach.* ACM Transactions on Graphics 11 (1992): 92-99

[5] Bederson B., Schneiderman, B., & Wattenberg, M.. *Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies.* ACM Transactions on Graphics (TOG) 21 (October 2002): 833-854

[6] North, Chris. *Information Visualization.* Proceedings of the IEEE Symposium on Information Visualization (InfoVis'02) (1992): 15

[7] Schneiderman B. & Wattenberg, M.. *Ordered Treemap Layouts*. Information Visualization, 2001. INFOVIS 2001. IEEE Symposium on (2001): 73-78

[8] Van Wijk, J., & Van de Wetering, H.. *Cushion Treemaps*: *Visualization of Hierarchical Information*. Proceedings of the IEEE Symposium on Information Visualization (INFOVIS'99) (1999): 73-78

[9] Vernier, F., & Nigay, L.. *Modifiable Treemaps Containing Variable-Shaped Units* (2000)

[10] Balzer, M., & Deussen, O.. *Voronoi Treemaps*: Proceedings of the 2005 ACM symposium on Software visualization (2005): 165-172

[11] Lu, H., & Fogarty, J**.** *Cascaded treemaps: examining the visibility and stability of structure in treemaps*. Proceedings of graphics interface 2008 (2008) : 259-266

[12] Bladh, T., Carr, D., & Scholl, J.. *Extending Tree-Maps to Three Dimensions: A Comparative Study*. Proceedings of the 6th Asia-Pacific Conference on Computer-Human Interaction (APCHI 2004) (2004): 50-59

[13] Bruls, M., Huizing, K. & van Wijk J.. *Squarified Treemaps.* In Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization. (1999): 33-42

[14] *Issue tracking system – Wikipedia, the free encyclopedia*. Retrieved September 11, 2009 from http://en.wikipedia.org/wiki/Issue_tracking_system

[15] *What is database management system?* – A Word Definition From the Webopedia Computer Dictionary. Retrieved September 11, 2009 from http://www.webopedia.com/TERM/D/database_management_system_DBMS.html

[16] *Scalable Vector Graphics (SVG)*. Retrieved January 27, 2010 from http://www.w3.org/Graphics/SVG/

[17] *Welcome to CodeIgniter : CodeIgniter User Guide*. Retrieved February 20, 2010 from http://codeigniter.com/user_guide/

X. Appendix

Appendix A: Class Diagram of TERMS

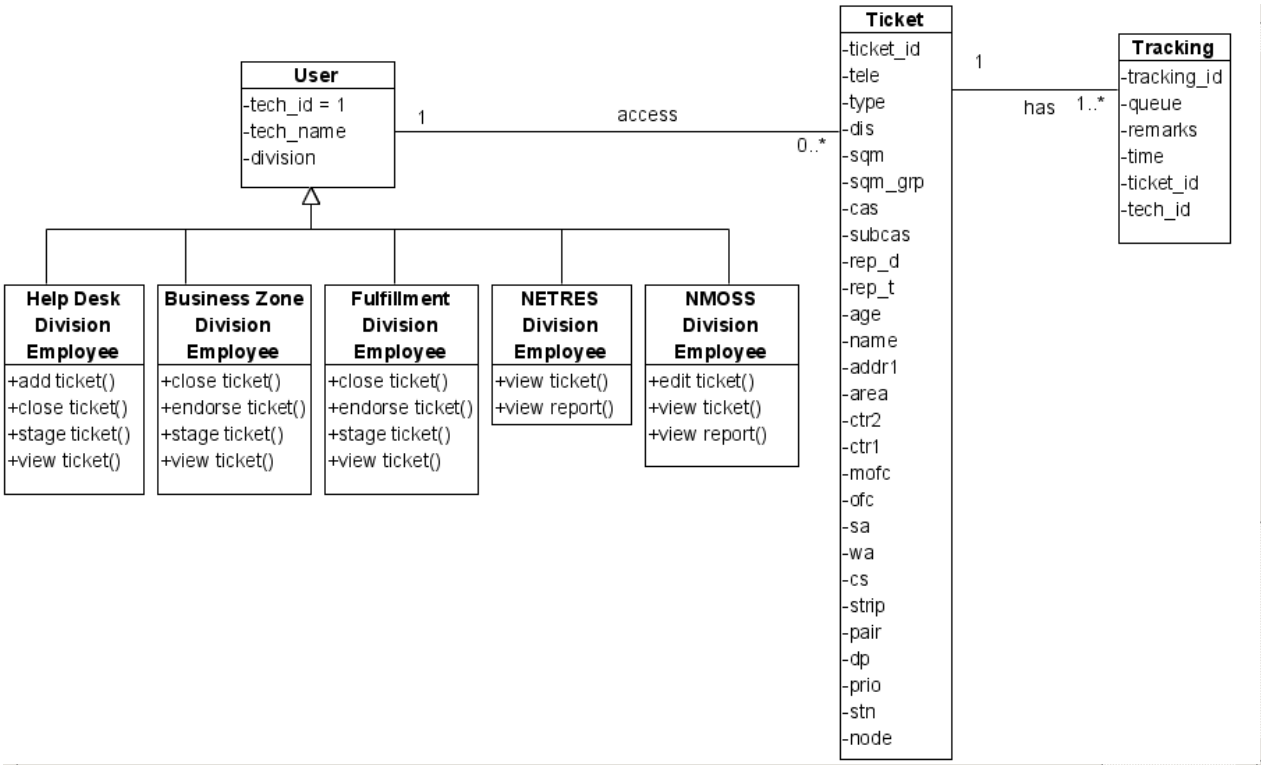


Figure 66. Class Diagram of TERMS

Appendix B: Object Diagram of TERMS

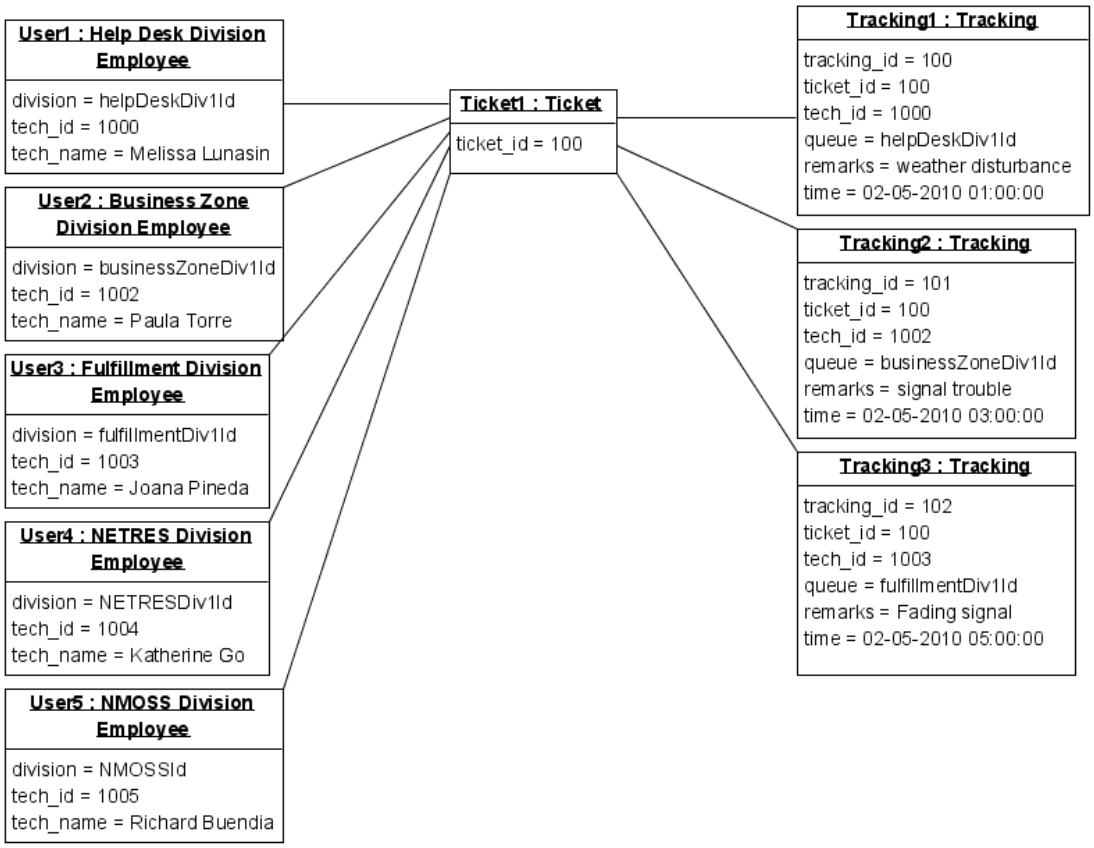


Figure 67. Object Diagram of TERMS

Appendix C: CodeIgniter Framework Architecture

CodeIgniter is based on the Model-View-Controller development pattern. MVC is a software approach that separates application logic from presentation. In practice, it permits your web pages to contain minimal scripting since the presentation is separate from the PHP scripting. [17]

- The **Model** represents your data structures. Typically your model classes will contain functions that help you retrieve, insert, and update information in your database.[17]

- The **View** is the information that is being presented to a user. A View will normally be a web page, but in CodeIgniter, a view can also be a page fragment like a header or footer. It can also be an RSS page, or any other type of "page".[17]
- The **Controller** serves as an *intermediary* between the Model, the View, and any other resources needed to process the HTTP request and generate a web page.[17]

CodeIgniter has a fairly loose approach to MVC since Models are not required. If you don't need the added separation, or find that maintaining models requires more complexity than you want, you can ignore them and build your application minimally using Controllers and Views. CodeIgniter also enables you to incorporate your own existing scripts, or even develop core libraries for the system, enabling you to work in a way that makes the most sense to you. [17]

The following graphic illustrates how data flows throughout the system:

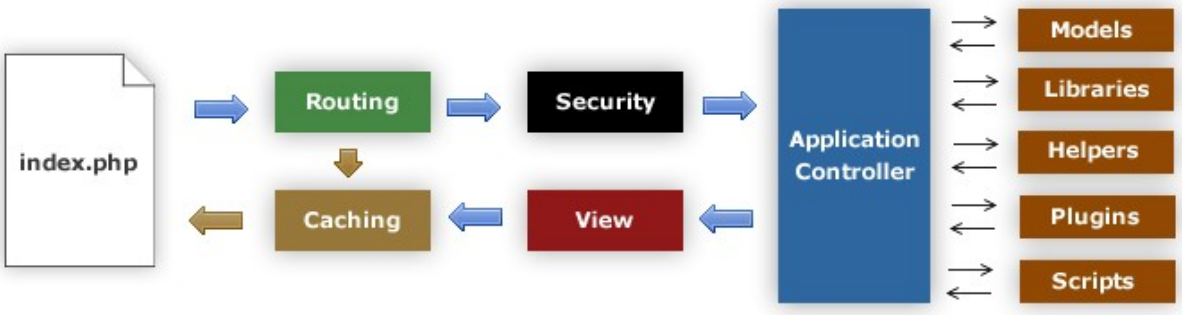


Figure 68. Data flow graph of the CodeIgniter Framework

1. The index.php serves as the front controller, initializing the base resources needed to run CodeIgniter.
2. The Router examines the HTTP request to determine what should be done with it.

3. If a cache file exists, it is sent directly to the browser, bypassing the normal system execution.
4. Security. Before the application controller is loaded, the HTTP request and any user submitted data is filtered for security.
5. The Controller loads the model, core libraries, plugins, helpers, and any other resources needed to process the specific request.
6. The finalized View is rendered then sent to the web browser to be seen. If caching is enabled, the view is cached first so that on subsequent requests it can be served.[17]

The CodeIgniter User Guide, found at http://codeigniter.com/user_guide/ houses the general features and conventions of CodeIgniter while the CodeIgniter Wiki, found at http://codeigniter.com/wiki/ contains regularly updated articles about the framework.

Appendix D. Source code

XI. Acknowledgment

I would like to thank God for giving me the strength, courage, endurance and confidence to finish my thesis. If it were not for Him, I wouldn't even be able to acquire a lot of skills, knowledge and experience. And most especially, I would not have met a lot of wonderful people.

I would like to thank my parents and sisters for their unwavering support and encouragement while I was doing my thesis. I'm most grateful for my sisters who were always there to remind and encourage me to achieve this goal. I would also like to especially mention my eldest sister Jody Ann Lunasin for being my primary resource person. Thank you very much for your patience and generosity.

I would also like to thank my adviser Mr. Richard Bryann Chua for being patient, considerate and helpful to me. Thank you for introducing the Treemap Algorithm, which I have successfully implemented in my system. This study would not have been possible without your help. Thank you very much.

I would also like to acknowledge all the Computer Science teachers from whom I have learned a lot, Sir Chua, Sir Zamora, Sir Co, Sir Gonzaga, Mam Lisa, Sir Patiňo, Sir Solano, Doc Magboo, Mam Sheila, Sir Tuaňo, Sir Dumanon and Sir Baes. We are very thankful for your admirable dedication and perseverance in the field of teaching. In behalf of all Computer Science students whose lives you have touched, I would like thank you for sharing with us your precious knowledge and skills.

Kim Carl Filomeno. Thank you for always supporting me, lending an ear when I need someone to talk to, pushing me to do my best and to stay focused and forcing me to concentrate when I lose sight of my goals. Thank you for always putting up with my stressful and depressed self and accepting me for who I am.

Katherine Go. Thank you Kat for the wonderful times we spent together at your house celebrating Christmas, making MPs or just lazing around at vacant hours. Thank you for occasionally lending us your neatly-written ComSci reviewers, we deeply appreciate it. Thanks for being the kindest and nicest girl in the group.

Paula Bianca Torre. Thank you for sticking with the Statistical Computing Track and enduring my childish teasing. Thanks for sharing your interests and movies with me and letting me tag along. Thank you for being honest though blunt with me all the time. Thanks for being the coolest girl in the group.

Richard John Buendia. Thank you for being a very funny and lively friend. I really enjoyed teasing you all the time and just making fun of anything under the sun. Thanks for being the funniest and most responsible guy in the group.

Jhamie Acebuche. Thank you for being at my company always and trusting me with your secrets and problems. I really appreciate how you make me smile from all your wonderful stories

and expressions. Thank you for being patient with me when sometimes I get really annoyed and being very honest with me. Thanks for being the bubbliest girl in the group.

Frances Joana Pineda. Thank you for making us laugh everyday without even trying hard. Thank you for sharing with me your hilarious stories and experiences and for being the funniest girl in the group.

I would also like to thank Perl and Eden for being very nice to me. I would like to thank Eden for sharing your interests with me, listening to my problems, giving me advice and allowing me to tag along during my free time. Thank you, Perl, for being such a nice and thoughtful friend and enjoying my company.

Finally, I would also like to thank everyone from BS ComSci Block 12 batch 2005 and 2006 for being wonderful classmates. Thank you everyone.