Otto-von-Guericke University Magdeburg

Faculty of Computer Science

Masters Thesis

A strategy Towards Evaluating And Making Sense Of Students' Learning Engagements

Author:

Kalu Oji Kalu

January 9, 2023

Advisors:

Prof. Dr. rer. nat. habil. Gunter Saake
Department of Technical and Business Information Systems
Otto-von-Guericke Universität

M.Sc. Chukwuka Victor Obionwu
Department of Technical and Business Information Systems
Otto-von-Guericke University Magdeburg

Dr. Ing. David Broneske
Deutsches Zentrum für Hochschul- und Wissenschaftsforschung

Kalu, Kalu Oji:
A strategy Towards Evaluating And Making Sense Of Students' Learning Engagements
Masters Thesis, Otto-von-Guericke University Magdeburg, 2023.

Abstract

The incorporation of digital learning environments (DLEs) into educational institutions has become increasingly prevalent, leading to the generation of huge amounts of data from students' use of these systems. Learning analytics (LA) can be used to analyze, and evaluate this data to improve educational processes. This thesis focuses on one such DLE, the SQLValidator, which is a online tool for learning and practicing SQL-related tasks. The aim of this research was to create a learning analytics dashboard (LAD) that could be used by instructors to understand students' engagements, behaviors and improve their performance in the SQLValidator system.

A design-based research approach was employed, in which there was collaboration with teaching assistants to iteratively design, develop, and evaluate a prototype LAD. The process included analyzing the problem and exploring the requirements for the solution, conducting user testing to gather feedback on the prototype, as well as analyzing students' engagements and errors using the LAD. The design of the LAD was informed by existing literature on data visualization, learning analytics, and error analysis.

The final LAD was well received by the teaching assistants(TA), with the survey results indicating that the TAs found that the dashboard helped them understand student engagement in terms of pain points, submission behaviors and their relationship.

The research suggests that plenty room exists for improvements and suggestions for future work includes conducting a larger-scale study with more TAs and evaluating the effect of collaborative work on student performance.

Statement of Authorship

I hereby declare that I am the sole author of this Master Thesis and that I have not used any sources other than those listed in the bibliography and identified as references.
I further declare that I have not submitted this thesis at any other institution in order to obtain a degree.

Signature: ____________ Place: ____________ Date: ____________

Acknowledgments

I would like to express my deep appreciation to Prof. Dr. rer. nat. habil. Gunter Saake for giving me the opportunity to complete my master's thesis within the DBSE research group.

My sincere gratitude also goes to my advisors, Dr.-Ing. David Broneske and M.Sc. Chukwuka Victor Obionwu, for their invaluable guidance throughout the process.

Lastly, I would like to thank my family and friends for their support.

Contents

List of Figures

List of Tables

List of Acronyms

AJAX	Asynchronous JavaScript And XML
DBC	Database Concept
DBR	Design Based Research
DLE	Digital Learning Environments
DM	DeLone and McLean
HCLA	Human Centered Learning Analytics
HTML	HyperText Markup Language
IS	Information System
IT	Information Technology
ITS	Intelligent Tutoring System
LA	Learning Analytics
LAD	Learning Analytics Dashboard
MOOC	Massive Open Online Courses
OAS	Online Assessment System
OVGU	Otto-von-Guericke-University Magdeburg
PD	Participatory Design
SQL	Structured Query Language
TA	Teaching Assistant
UI	User Interface
UX	User Experience

1. Introduction

1.1 Motivation

There has been a consistent rise in the number of educational institutions incorporating online learning into their curriculum. This can be done to help to maintain an advantage and to make classes more accommodating to a increasing number of students.The use of online learning has also been seen to rise in times when physical learning is not possible (as was required during the COVID19 pandemic). As a result of these developments, teachers could be exposed to these digital platforms at some point in their teaching career [Clark-Ibáñez and Scott, 2008] .

One way whereby universities have taken their programs online is through implementing digital learning environments (DLEs) . These environments include learning platforms like simulation-based learning environments, Learning Management Systems (LMS), Intelligent Tutoring Systems (ITSs), Massive Open Online Courses (MOOCs), electronic lab modules and many others. Online learning platforms assist in simplifying and managing various aspects of e-learning such as registration, delivering courses, monitoring and reporting student progress. These platforms also generate a huge amount of data from student interactions with the system.

Learning analytics (LA) uses the data from online learning platforms to help institutions understand and enhance the educational processes. The data in question is generated from student actions, such as completing taking exams, assignments, through Online Assessment Systems (OAS), online social interactions, extracurricular activities,e.t.c. These data can also be used to map students' behaviours while learning [Blikstein, 2011]. This thesis will focus on an OAS called SQLValidator. It is a web-based platform that facilitates learning and practicing SQL related tasks. The system allows students to construct and test their queries against a database, and receive instant feedback. The detailed information about this system will be discussed further in the upcoming sections.

According to Johnson et al. [2011],one important utilization of LA is the ability to keep track of and anticipate students' performance in learning, to identify potential difficulties early on, and to provide appropriate interventions accordingly. This is particularly true for analyzing errors from assignments and exams because the more educators learn about the nature of students' challenges, the more effective educators can be [Spohrer and Soloway, 1986a].

As the data generated by LA systems increases, it becomes harder for an instructor to analyze data and keep track of the progress of students using traditional means. Hence, keeping track of patterns and analyzing students performance become difficult to in tables. This has fueled the development of learning analytics dashboards (LAD) which are used to assist teachers in getting an overview of students' activities, re-examine their teaching process, and to find where students require help [Verbert et al., 2013]. Through clear and relevant visual representation in LAD, key strategies and decision making can be carried out by instructors. Dashboard should contain limited, understandable information [Halim, 2021].

However, most current systems do not give instructors rich information or the level of interactivity they would like [Bodily et al., 2018]. To tackle this problem, the importance of targeting the design to people who would actually use these systems has been outlined [Shum, 2018].

Verbert et al. [2013] in their work proposed a model that suggests that in order to make use of data effectively, teachers (and students) should first identify their questions and evaluate how relevant the data is to answering those questions. Without this initial step, the data on its own may not be useful. Other researchers like [Bakharia et al., 2016] and [Li et al., 2021] have also highlighted the importance of gathering teachers' questions and connecting them to educational concepts, which would help data being reported back to actionable. Unfortunately, there seems to be a shortage of the research works which can guide how visual design of LA interfaces can be done using the relationships between educational constructs and teachers' needs .

In addition, [Xhakaj et al., 2016] has observed the insufficiency of error information in current dashboards, which makes it difficult for teachers to carry out comprehensive error analysis. While some studies have attempted to study students' programming errors, far less work has been conducted on SQL [Poulsen et al., 2020].

1.2 Aim

The goal of this thesis is to build an interactive teacher-centered LAD which would help teachers monitor and make sense of students' engagement with exercises in an SQL programming course. It would provide features to analyze different forms of errors committed by students. This system will be built upon an already existing SQLValidator which is currently used in the Computer Science department of OVGU.

The thesis will furthermore focus on the following research questions :

RQ 1 : How useful is the LAD in making sense of students' engagement with exercises.

RQ 2 : What are the common types of programming errors among students learning SQL.

RQ 3 : What SQL errors classes are the most severe for students.

RQ 4 : Is there any correlation between submission behavior and the frequency of errors committed by students learning SQL.

1.3 Outline

The structure of the thesis follows as so: Chapter two gives a background on the key concepts relating to Learning Analytics, Error Analysis and Visualization. Chapter three summarizes the related scientific literature on error analysis and visualizations in Programming related courses. Chapter four presents the methodology employed in the thesis. Chapter five outlines how the LAD was implemented . Chapter six provides the testing, evaluation of the LAD and analysis of errors of student errors in an SQL course. Finally, Chapter seven summarizes the important results as well as provides ideas for future work.

2. Background

This chapter provides a grounded understanding of the principles and technologies related to this work.

2.1 Learning Analytics

There have been several definitions of LA according to many researchers. [Chatti et al., 2012] describes it as the use of organized data, data produced during the student educational engagements and analytic models to uncover information, social associations and to predict and instruct on learning engagements.

Through the use of MOOCs, LMS, social media, and other OLEs, a lot of digital trails are generated from learning process. These digital trails and data points can grow to become large and can become difficult to interpret, thus needing additional effort in order to fully comprehend. The Learning Analytics Life Cycle (see Figure 2.1) introduced in by Khalil and Ebner [2015] consists of four parts;
- The learning context where data is generated.
- The data itself, which can consists of a variety of datasets.
- The analytics, consisting of various analytical methods.
- The Act, involves achieving objectives to improve the learning process.

2.1.1 Learning Analytics Process

LA is an iterative cycle, which is usually implemented out in three phases [Chatti et al., 2012]. The phases in Chatti et al. [2012] work include:
(A) Data acquisition and pre-processing
(B) Analytics and action
(C) Post-processing.

- *Data acquisition and pre-processing*: Data usually comes from various sources and can exist in different formats. These sources could include various educational

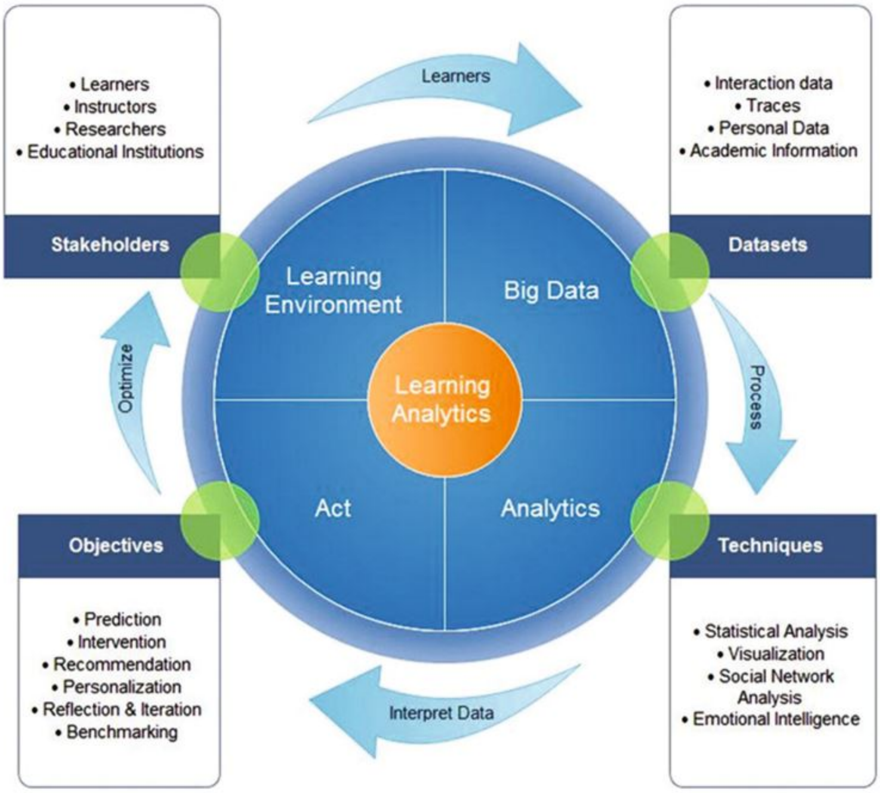

Figure 2.1: LA Life Cycle. Source, [Khalil and Ebner, 2015]

environments and systems such as LMS,OAS, MOOCs, etc where students carry out learning activities. It is generally required to carry out data aggregation and pre-processing, since the data collected might be too much or comprises unnecessary details. Additionally, it is a common practice to change the format of the data to fit a particular LA method.

- *Analytics And Action*: The following step involves utilizing LA methods to analyze the data in accordance to the user's objectives. Techniques such as visualization are frequently employed to assist users in comprehending the results of analytics more efficiently, especially when dealing with big data sets. The data can be examined to uncover underlying patterns. This can help support users in making decisions and taking actions, such as adapting, monitoring, assessing, analyzing, intervening, predicting, personalizing, reflecting and recommending.

- *Post-processing*: This stage is focused on ongoing improvement of the analytics project. It may involve gathering new data from other sources, changing variables, identifying new metrics, or choosing an entirely different analytics method.

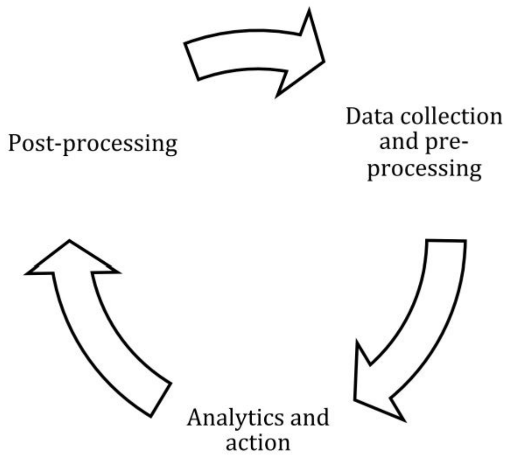

Figure 2.2: LA process. Source, [Chatti et al., 2012]

2.2 Visualization And Dashboards

As processes and institutions continue to generate data, the need for presenting these data in a way that is easy to comprehend and access becomes crucial [Sadiku et al., 2016]. According to Ajibade and Adediran [2016], data visualization refers to the presentation of data in a visual layout. Azzam et al. [2013] describes data visualization as a process that is hinged on quantitative or qualitative data which produces a visual representation of the data and also supports examination, communication and exploration of data. It relies on the design of effective and sometimes visually pleasing visual representations which users can manipulate for exploration or to solve particular tasks [Klerkx et al., 2014].

Few [2006] defines a dashboard as a display of very relevant information required to realize objectives; usually put together on a single screen to enhance easy monitoring.

2.2.1 Concerns Of Visualization

When using data visualization, there are quite a number of challenges or cautions to be aware of. Key concerns include:

- It can sometimes be difficult to decide what visual might be the best to represent your data [Sadiku et al., 2016].
- Not to visualize all the info or datasets on the same graph [Eckerson, 2013].
- Prioritize Simplicity [Eckerson, 2013].
- Dashboards should have intuitive, concise and clear display mechanisms [Few, 2006].
- Dashboards should show information relevant to the realization of specific goals [Few, 2006].
- Dashboards should be customizable.

2.2.2 Visualization techniques

Many factors affect the selection of suitable visualization techniques, such factors include; goals of the user, amount of data, and the variables/indicators to be represented.

In addition to standard techniques shown in table 2.1, there are other, more sophisticated techniques. They include ; stacked, iconic, dense pixel and geometrically transformed displays [Keim, 2002], which are beyond the scope of this thesis.

2.2.3 Design Considerations for Dashboards

Several authors have analyzed dashboards and have come up with recommendations for good design principles for dashboards. Some of those recommendations include:

- Choose the right layout [Eckerson, 2013].
- Not to visualize all the info or datasets on the same graph [Eckerson, 2013].
- Prioritize Simplicity [Eckerson, 2013].
- Provide more data context without cluttering up the visualization [Eckerson, 2013].
- Select the correct type of charts/visualization technique [Eckerson, 2013].
- Possibility to observe the information at a glance [Few, 2006].
- A dashboard should be fitted into a single screen [Few, 2006].
- Dashboards should have intuitive and clear display mechanisms [Few, 2006].
- Dashboards should show information relevant to achieving specific goals [Few, 2006].
- Dashboards should be customizable.

[Few, 2006] pointed out thirteen greatest mistakes one can make while creating dashboards. Those mistakes can be summarized as:

- Omitting key information.
- Using too many different colors.
- Using multiple screens.
- Cluttered display.

While designing a dashboard, these recommendations should be taken into consideration as much as possible.

Visualization Technique	Description	Attributes
Pie Chart	Circular graphic representation of categories where each slice shows percentage or numerical proportion.	**When to use :** a. Comparing parts that add up to a whole. **When not to use :** a. with data partitions that do not add up to a whole. b. Comparing a large number of variables. [Gupta, 2019]
Box plot	A graphical method of displaying skewness and distribution of numerical data.	**When to use :** a.Displaying data distribution through percentile. **When not to use :** a. Analyzing individual datasets.[Gupta, 2019]
Scatter Plot	This uses data points on a vertical and horizontal axis to show the strength of the relationship between two numeric variables. [Soma, 2016]	**When to use :** a. General overview of variables. b. Tracking progress of a few variables. **When not to use :** a. One-dimensional data. b. categorical/non-numeric data. [Gupta, 2019]
Bar Chart	This chart is used to compare quantities of different categories with rectangular bars of heights representing numerical values.	**When to use :** a. Comparing a few variables in the same category. b.Tracking progress of a few variables. [Soma, 2016] **When not to use :** a. Visualizing continuous data.
Line Chart	This chart is used to indicate trends and progress of variables over time [can, 2022].	**When to use :** a. Trend and time series analysis. b.Comparing several variables over time. c. Predictive analysis. **When not to use :** a. Analyzing individual components or sections. [Gupta, 2019]

Table 2.1: Standard Visualization techniques.

2.3 Students' Programming Errors

Programming is a basic skill needed by students of computer science to effectively write and debug computer programs. Most students who begin a computer science program are usually novices and hence usually make errors while learning to write programs. Veerasamy et al. [2016] describes an error as a flaw in source code that leads to failure in compilation or running of the program. It is important for teachers to identify errors and misconceptions of students, so they can devise approaches to address them, and improve their teaching quality and methods [Veerasamy et al., 2016].

To help programmers identify errors, programming language compilers typically tend to produce diagnostic messages, which are commonly referred to as error messages [Mccall, 2016]. In education, compilers can be used to highlight students' errors to help students debug, and at times these errors are checked by teachers to analyze and help improve students' programming skills ([Spohrer and Soloway, 1986b] ; [Bringula et al., 2012]).

According to Veerasamy et al. [2016], a novice programming student would usually fall into several types of errors because of misconception, misjudgment, poor attention, habits (e.g procrastination), etc.

2.3.1 Error Categorization

Since the dashboard was built on-top of the SQLValidator, the error categorization already produced from the SQLValidator was used. Obionwu et al. [2021] grouped the errors into groups called error classes. The error classes were gotten from the error codes used in the work of Obionwu et al. [2021]. The table below shows the error classes and the kind of error they represent.

2.3.2 Procrastination

Procrastination can be described as a failure to properly manage time. This behavior has been observed to lead to poorer outcomes for students [Park et al., 2018]. According to Ferrari et al. [1995], this can be apparent when students delay to carry out on formal types of tasks related to their academic work. Procrastination in students can be seen in various tasks, ranging from writing assignments to doing administrative tasks.

A common method to quantify students' procrastination is by calculating the timing of students with regards to their interaction in an LMS prior to an important deadline [Park et al., 2018].

Entezari et al. [2018] addressed three measures for characterizing students' behaviors in submission of homework and tasks. These measures include :

- **Start Gap** : This measures how early a student begins to solve a task. This is calculated as the difference in time between the task submission deadline and the first pen stroke (in this case the first submission attempt) of the task. This entails that students that procrastinate would usually record a smaller start gap than others [Entezari et al., 2018].

Error Class	Tag	Description
Syntax		
0	Syntax Error	Error triggered when the query submitted is not syntactically correct.
Table		
2	Column Count Error	Error triggered when the submitted query fails to return the expected number of columns.
3	Column Order Error	Error triggered when the returned column order does not match the expected order.
4	Column Name Error	This is triggered when the column cannot be found in the table.
6	Table Row Count Error	Error triggered when the submitted query does not return the expected number of records.
16	Table Name Error	triggered when the table name in the submitted query is not expected table name.
20	Table Content Error	Error triggered when the submitted query return the correct number of records but the records do not match expected records.
21	Table Row Order Error	Error triggered when the records returned do not match the expected order.
Foreign Keys		
17	FK Name Error	Error triggered when the foreign key name in the query submitted does not appear in the table.
18	FKRef Table Error	Error triggered when the table in the reference constraint does not appear in the database.
19	FKRef Column Error	Error triggered when the referenced table does not contain the column in the foreign key.
Constraints		
7	Constraint Count Error	Error triggered when the number of constraints is not the expected number.
8	Primary Key Error	Error triggered when the the primary key in the submitted query is not expected primary key.
9	Unique Key Error	Error triggered when the the unique key in the submitted query is not the expected unique key.
10	Foreign Key Error	triggered when the foreign key in the submitted query is not the expected one.
11	General Keys Error	triggered when the general keys in the submitted query is not the expected result.
Schema		
5	Column Count CT	triggered when the number of columns is not the expected number during table definition.
12	Data Type Error	triggered when the data types of columns are not the expected types during table definition.
13	IsNull Error	triggered when the isNull constraint is applied to wrong column during table definition.
14	IS DEFAULT Error	triggered when the IS DEFAULT constraint is applied to wrong column during table definition.
15	Column Name Error	triggered when any of the column names during table definition is not the expected one.

Table 2.2: Error Classes and Descriptions adapted from [Obionwu et al., 2021].

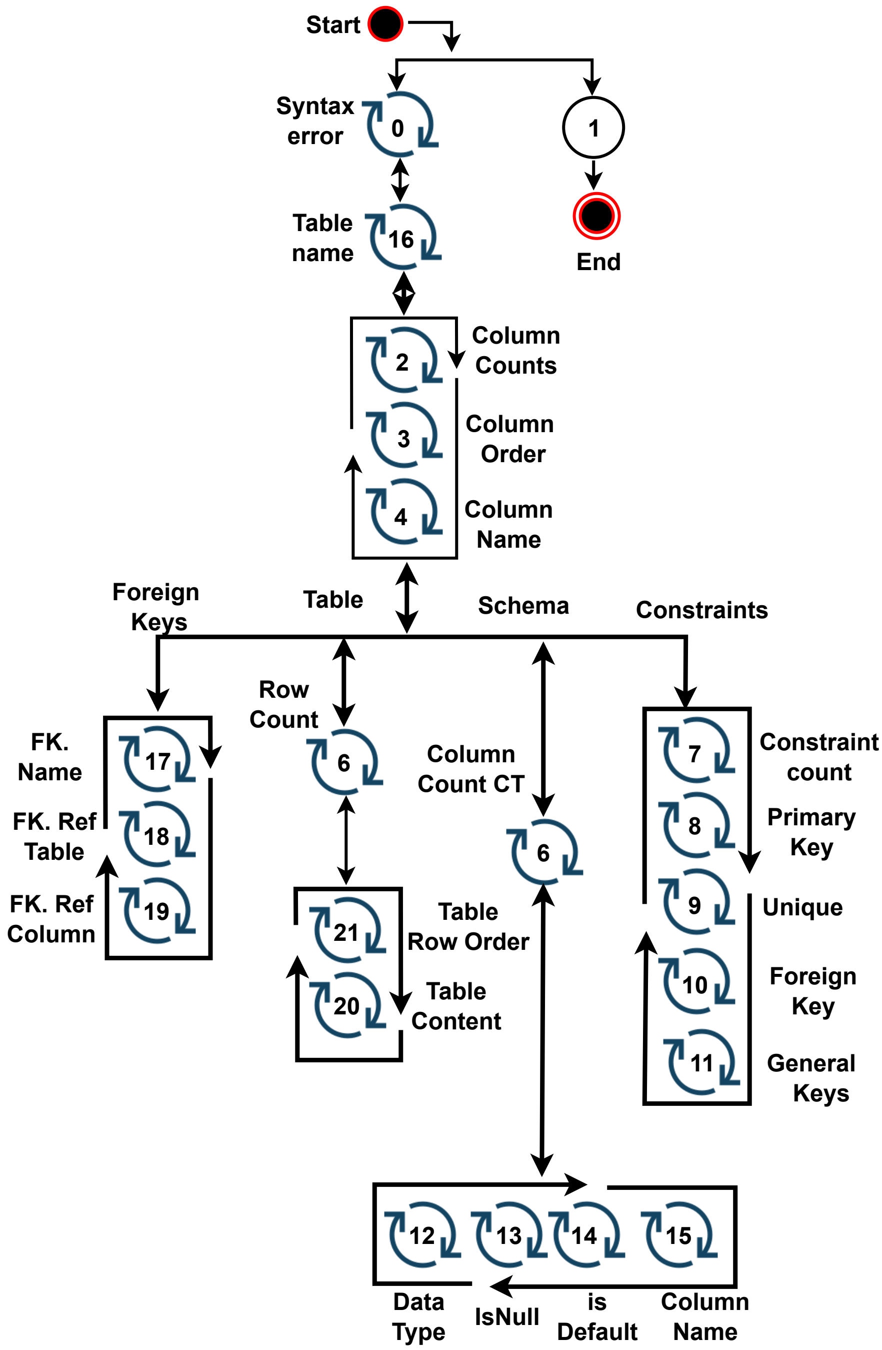

Figure 2.3: Error Classes. Source, [Obionwu et al., 2021]

- **End/Finish Gap** : This metric quantifies how late a student completes a task. It is computed as the time gap between the time of the final input (in this case the final recorded submission for the task) and the task's submission deadline. Similar to the start gap, a shorter Finish Gap could indicate procrastination.

  The final finish gap is calculated as the average finish gap over all tasks given in an assignment [Entezari et al., 2018].

- **Effort Density** : This metric determines how long it takes to finish a task in relation to the overall time. It is calculated by finding the ratio of the time spent solving the task to the total time elapsed from the first input to the final input (submission) .

  This means that students who take over several days to complete an assignment will have a lower value of effort density, while a student who completes an assignment in one sprint or session will have a high value approaching 1 [Entezari et al., 2018].

  This measure can be high for students who procrastinate, as they typically begin late and try to finish the assignment as quickly as possible or in one problem-solving session. However, it's not always the case, as a student who starts early can also choose to finish the assignment quickly, leading to high effort density values [Entezari et al., 2018].

$$d_k = \frac{W_k}{T_k}$$

  Where, W_k is the total active time spent on the task k, and T_k is the total time from the first input to the submission/last input on the task.

  The final effort density 2.1 is the average effort density over all tasks given in an assignment, where n being the number of tasks.

$$D = \frac{\Sigma_{k=1}^{n} d_k}{n} \tag{2.1}$$

2.4 Human Centered Learning Analytics (HCLA)

HCLA is a field of LA research that examines the human related factors that can affect the efficient utilization of LA tools. It also explores the use of co-design and participatory design to develop more useful LA innovations.[Buckingham Shum et al., 2019].

Recently, the adoption of human-centred design principles has increased as stakeholders are being involved in the creation process LA tools. [Ahn et al., 2019] emphasized the importance of human-centered design in his work. He stressed on the need to take into account the requirements and perspectives of all interested parties involved as part of the design of LA interfaces. This would guarantee effective integration of the technology and educational practices. Figure 2.4 gives a picture of the present situation of the human-centred design landscape.

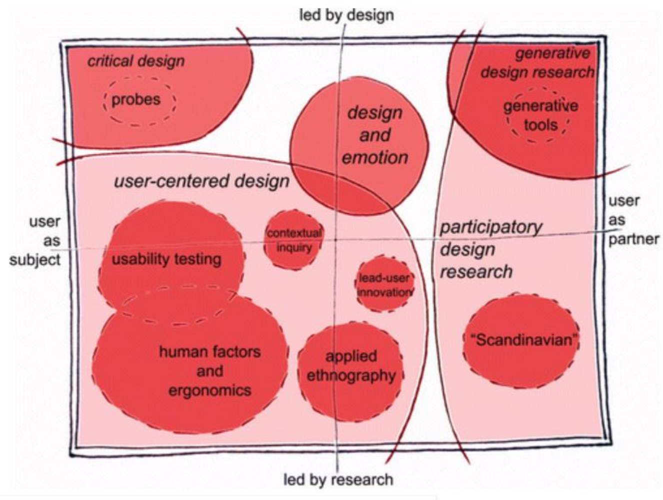

Figure 2.4: current state of the human-centred design. Source, [Sanders and Stappers, 2008]

2.4.1 Co-Design In Learning Analytics

Participatory design(PD) was first defined by [Asaro, 2000] as a strategy for actively engaging stakeholders in an artifact's design process, to the end that the outcome of the technological development or artifact meets the design requirements. According to [Sanders and Stappers, 2008] co-design refers to a core strategy within PD where collaboration becomes an expression of collective creativity between the designer and stakeholders. In some cases, user centered design,PD and co-design are used interchangeably.

Co-Design Process

While there is an increasing need for co-design in LA, the adoption of co-design approaches has rather been slow [Dollinger and Lodge, 2018]. Furthermore, how co-design principles are being adopted in LA is also not well known [Sarmiento and Wise, 2022].

Sarmiento and Wise [2022] in their work, investigated in what ways co-design has been applied to LA by searching and grouping different approaches from 90 research studies involving co-design in LA. Their work tried to address the questions :
(1) Which stakeholders are involved, how to select them?
(2) What design tools and techniques are utilized, and at what stage are they necessary.

The design tools and techniques realized from [Sarmiento and Wise, 2022] can be grouped into the following categories:

- *PD for comprehending needs before design* : This includes certain techniques used to gain understanding of the workflow and needs of stakeholders. Of all the techniques investigated, collecting and assessing needs/requirements were by far the most frequently employed. Some of these requirements were got by prompting teachers to think about the kind of superpowers they thought could assist their work [Holstein et al., 2019]. Other designers performed needs assessment by taking on the persona of "learners" while the stakeholders became "experts" so as to understand them more.

- *PD during design*:While taking into consideration stakeholders requirements and needs are important, co-designing techniques which involve bringing stakeholders' into the design itself are what constitute the meat of PD.

  Brainstorming with the stakeholders is one of the techniques involved in this category. This is especially advantageous when the researchers lack certain knowledge which the stakeholders have.

  Another strategy is to combine designer's expertise with the stakeholders' insights . This could be achieved through interviewing stakeholders then using that gathered knowledge to create a mood board with alternative designs which then is the utilized by the researcher(or designer).

  In addition, artifacts such as cards or some sort of game is used to elicit the stakeholders' knowledge.An example of this can be found in [Brun et al., 2019], where cards that show parts of a LA tool were given to stakeholders in an exercise for them to mix into a dashboard.

  Furthermore, Early evaluation of prototypes or design concepts using feedback from stakeholders. This could involve showing different solutions for an LA tool to stakeholders or having them use an early iteration of a prototype of the intended LA tool.

- *PD for testing after development* : Sarmiento and Wise [2022] observed that researchers tested a prototype with participants either in more controlled environments such as in a lab or in real classrooms.

  The researchers usually partner with instructors to test an initial version and then get feedback from the instructors which would inform subsequent updates to the design. The tests proved important as they helped researchers see many situations they had not envisioned while designing the tool.

- *Co-Development* :Sarmiento and Wise [2022] discovered that some other projects in their study took a different approach different from the three earlier mentioned approaches. In this approach the researchers/developers added incremental enhancements or totally new features by having constant interaction with stakeholders over the course of development [Fischer et al., 2009].

Liaqat et al. [2018] in their work outlined the various advantages that arise from PD practices. Such advantages include :

1. Exposes hidden assumptions : PD can help provide additional insight, which may highlight inconsistencies and incorrect assumptions on the part of designers.
2. Provides context to empirical findings : Designers often realise certain patterns or requirements from their personal inquiry into the needs of the end users. But when PD is involved, these patterns start to make sense and provide designers and researchers with deeper understanding of these needs.

3. ldentifies limitations of existing technology : The needs of users might be very complex which would need going beyond the existing technology and thus, designers may need to create new ways to meet them.
4. Brings theory into the real world : PD when well implemented can extend principles from educational research into useful resources which can be incorporated into the design of relevant technologies.

3. Related work

This chapter is focused on giving an overview of several recent works done on the subject of LAD and error analysis of students' assignment.

3.1 LA Dashboard

[Park and Jo, 2019], Corrin et al. [2016] and other authors have brought to question the visual considerations of dashboard design, primarily focusing on the how complex the visual elements and data are.

Fernandez Nieto et al. [2022] in their work compared three alternative designs to visualizing student data. The goal of the work was to discover what were teachers' perceptions of visual-narrative designs and their intended uses of the designs to support their work.

They interviewed four teachers individually with each interview spanning forty five minutes. These interviews were then recorded in video format,transcribed and coded using a software called NVivo. They analyzed the teachers words and actions while exploring the designed interfaces. The study found that different forms of data visualization offer unique insights into the processes of education. Specific methods of displaying information can reveal certain patterns while hiding others. Therefore, it is crucial for scholars to further investigate the effects of various visual storytelling methods and the selection of data visualization techniques on instructional practices and student learning.

In most DLEs there is need for teachers to know how students are performing based on continuous assignments solutions. In their work, [Halim, 2021] created a real-time dashboard to monitor student activities regarding attendance and continuous assessment. This tool was made available to both teachers and students, to monitor students' attendance to classes and their continuous assessment performance. The dashboard consisted of three parts; student information, student continuous assessment performance and the attendance report.

3.1.1 User-Centered LA Dashboards

While there is no one way of designing user-centered dashboards, many researchers have come up with several processes which incorporate the stakeholders' input into the design of LAD's.

Demmans Epp et al. [2019] created a visualization tool which would present students' analytics within a particular Online Learning Environment(OLE), called PeppeR to the teachers. The goal was to reformulate the students' activity reporting features, due to the limitations that were experienced in PeppeR, in order to make data more available and useful to the instructors. The work combined two iterative practices; design-based research and user-centered design.The work did not report on assignments or exercises done by students.

The method employed by Demmans Epp et al. [2019] involved a 3 - step iterative process or phases. This design process only involved the end users in one step of the iteration and was not evaluated by the users(but by visualization researchers) before integration into the system.

Another work by Ez-Zaouia [2020] applied principles of Human Computer Interaction (HCI) and InfoVis.The approach employed in this study involved the application of a process model informed by Munzner [2009]. The proposed model aim to use visualization techniques to address specific areas of concern and they accomplish this by utilizing four levels of integration. This process model serves as a guide to organize the development of dashboards, to inform decisions made early on, and to document progress throughout the design process.

Recently, educators have been utilizing a combination of audio-visual and text-based platforms in addition to the Learning Management Systems (LMS) provided by their educational institutions..This motivated Pozdniakov et al. [2022] to design and deploy a tool for real-time monitoring of students when learning and working in teams. Their study was divided the design and validation phases. The second part of the Pozdniakov et al. [2022] study design was deployed and validated by usage by two TAs in an authentic university context.

3.2 Students' Programming Error Analysis

There have been many works which have analyzed errors that novice student programmers make while learning imperative or functional programming languages, but far less research has been conducted on SQL [Poulsen et al., 2020].

Altadmri and Brown [2015] in their work, analyzed the frequencies of several programming errors and the time students take to correct them from 37 million compilations. In addition, they found that teachers' initial beliefs about error frequencies did not tally with the data.

Tabanao et al. [2011] used Jadud's Error Quotient and compile-time errors and to detect struggling students who were at risk of fail the course. Jadud [2006] described the error quotient as an algorithm which finds the correlation between students' errors and their performance. In their work, they also predicted midterm exam scores.

When it comes to SQL, [Taipalus and Perälä, 2019] tried to understand types of errors that are persistent and which types of errors are associated with the different SQL sublanguage features. They found that the most persistent errors are the logical errors.

Ahadi et al. [2016] performed quantitative analysis of syntactic mistakes made by students learning SQL. They considered 160,000 SQL queries from 2000 students across 8 years. They also developed an automatic classifier for predicting students' performance. According to Poulsen et al. [2020], many studies have focused on analyzing the results of learners' summative evaluation in SQL, but there is a lack of research on the challenges students encounter while completing SQL homework assignments. However, this gap in the literature is only recently starting to be addressed. Taipalus and Perälä [2019].

4. Method

In this chapter, I present the methodology employed in the design of the LAD. The iterative design-based research approach was employed and dashboard designed guidelines were followed [Few, 2006]. The qualitative components of the work were reported in compliance with the standards for qualitative research reporting.

4.1 Context

The exercises considered by this study were from students undertaking an introductory course to databases and SQL.

4.2 Design Based Research (DBR)

There have been many works which have described DBR. Wang and Hannafin [2005] described DBR as a methodology that is both adaptable and structured, which aims to improve educational practices by using an iterative process of evaluation, formulation, development and implementation, through teamwork between stakeholders and researchers.

Choosing an iterative design process which involves designing prototypes, mock-ups,etc helps create a continuous feedback loop. This early feedback can provide the designer with the needed information so as to create highly effective artifacts [Prieto Alvarez, 2020].

According to [McKenney and Reeves, 2018], DBR comprises of four phases:

1. Analysis: Analysing the problem and exploring the scope and requirements for the needed solution.
2. Design and Construction: Designing and developing the solution from knowledge based on existing literature.
3. Evaluation and Reflection: The iterative cycles of evaluating and identifying of opportunities to refine the solution.

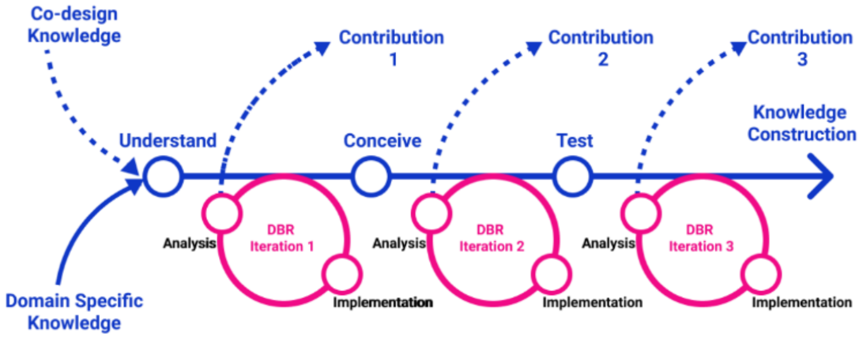

Figure 4.1: DBR phases using co-design, [Prieto Alvarez, 2020]

4. Contribution: Advancing theoretical and/or practical knowledge .

The application of PD or co-design principles in each iteration of the DBR can be visualized in Figure 4.1 [Prieto Alvarez, 2020].

Phase 1. Analysis and exploration

Various literature on learning analytics([Eckerson, 2013],[Few, 2006]), data visualization, and error analysis were reviewed to provide context and guidelines required to support the study. An interview was carried out with two teaching assistants in the university to understand their needs and expected outcomes from the dashboard.

Phase 2. Design and construction

The first interview from the previous phase was transcribed and qualitatively analyzed to give information for the creation of 2 mock-up designs using Figma. Another meeting with the TAs led to the selection of one design which was translated to the first prototype.

Phase 3. Evaluation and reflection

The prototype from phase 2 was presented to the TAs in a subsequent meeting to gather their feedback. During the interviews, the TAs identified through testing of the dashboard, what things they wanted modified, added or removed. Phases 2 and 3 alternated (Monthly for 2 months) throughout the remaining parts of the project timeline with each feedback informing design changes.

Phase 4. Implementation and spread

After the last iteration from phase 3, the dashboard was presented to the TAs and is planned to be incorporated into the SQLValidator System used in introductory course to SQL programming.

4.3 Qualitative analysis

Qualitative analysis was conducted on data gotten from interviews using Thematic analysis [Braun and Clarke, 2012]. Analysis was done to identify the teachers' needs and the elements (e.g. data and visualizations) needed to meet them.

As data was gathered, a codebook was created and representative quotes were recorded for each code. The codes were grouped into categories based on the teachers' needs and the dashboard elements that addressed them. The codes and categories were updated as more data was collected. These groupings were then used to create themes. The feedback from the teachers were also taken into account and any changes made were tracked. Table 4.1 shows the Themes which represent teachers needs and the dashboard elements used to address those needs.

4.4 Data Source

The data used comes from students' exercise submissions of an introductory course to databases and SQL in two semesters; Winter semester of 2021(WS2021) and summer semester of 2020 (SS2020) . Between the above mentioned semesters, about 310 students attended the course. The take-home exercises attached to the course majorly tested students knowledge on SQL programming using an automated assestment and feedback system called SQLValidator [Obionwu et al., 2021]. The exercise covered several language featuressuch as ; the Data Definition Language (DDL), the Data Manipulation Language (DML), and the Query language (QL).

Majority of students (above 70%) who attended this course are novices in SQL programming as ascertained by a survey taken at the beginning of each semester.Students submitted their solutions to SqlValidator. Which in-turn analyses the solution and saves each solution with the classes of errors committed and the time of submission.

4.5 Dashboard programming

The dashboard runs on a centralized architecture made up of three parts: A database system to store student logs, a back-end server to perform queries on the database and the client to take user inputs and display information.

The dashboard functions by generating visualizations in real-time using data from the backend server. The visualizations are created using ChartJS, a Javascript charting library, allowing for an interactive and responsive experience for the CC members across various screen sizes and orientations. As part of the SqlValidator, only administrators and teachers who have admin privileges can access the dashboard.

4.6 Error Analysis

4.6.1 Error Frequency

This refers to the total count of occurrences of a certain error class. An error can occur multiple times in the series of submissions targeted at solving a particular task. To calculate the frequency of an error class, each occurrence is counted per submission if it appears in subsequent submissions.

Theme	Teacher Need, Dashboard Elements and Quotation
1. Explore Students' Assessment Data	
1.1. Quantitative Error Metrics	TA1 : "We want to see the number of errors for each task, so we can know where to focus our efforts" **Addressed with bar chart** Figure 5.9 TA1 : "Students who consistently experience errors can be struggling with the course, so we need to identify them" **Addressed with drop down to select individual students** TA2 : "There are different error classes which are recorded by the SqlValidator from student submissions, it will be nice to know which ones are most prevalent" **Addressed with bar chart** Figure 5.8
1.2. Students' Engagement Behaviours	TA1 : " So we know you see the trend of the submissions and that can actually give you a behavior. Yeah, so if you have this particular visualization, we can now use it to track procrastination." **Addressed with scatter plot** Figure 5.16 TA2 : " You know sometimes a student might not be really engaged with the exercise, so we can tell whether the errors may be as a result of a behaviour trend or just lack of knowledge."
2. Explore Students' Questionnaire Responses	
2.1. Student Demographics Figure	TA1 : " Students fill out a questionnaire as the beginning of the semester, we can see the students' demographics when we open the database" **Addressed with scatter plot** Figure 5.7
2.2. Students' Self Assessment Figure 5.12	TA1 : " Check the database for the questionnaire responses and make charts to visualize how the students answer" **Addressed with scatter plot** 5.12
3. Contextualization of Analysis Addressed with Figure 5.12	
3.1. By Semester Figure	TA2 : "We can have a drop down to change the semester for the charts"
3.2. By Exercise Groups	TA2 : "Can we filter the results by exercise groups, there are different exercise groups students belong to in the course"
3.4. By Proficiency	TA1 : "I will like to see how students who consider themselves proficient in SQL would perform in the exercises. We need to compare them with the novices"
3.4. By Students	TA1 : "We can select the best performing student from one exercise group and compare to the worst student in that group"

Table 4.1: Thematic analysis of teachers' needs and the dashboard elements used to address them.

4.6.2 Error Duration

The error duration can be described as the average time required for a student to fix a particular error also called `time-to-fix` [Albrecht, 2022]. Different researchers have tackled the calculation differently.for example, Mccall [2016] found the average time needed to submit a solution without a particular error after the error has been encountered. The maximum time given to fix an error is 5 minutes as any time longer than that is assumed to be idle time. Albrecht [2022] computed error duration by calculating the difference in number of submissions from when an error is spotted to when it is resolved. In this thesis, I will be employing the method used by Albrecht [2022].

4.6.3 Error Severity

According to Mccall [2016], error severity aims to identify which aspects of a programming language students have the most difficulty with. It measures the average effort required to fix a specific error and it is computed by multiplying the frequency of the error and the duration of the error.

4.7 Dashboard Evaluation

For the evaluation phase, 3 teaching assistants who were not part of the LAD development phase were invited via Zoom to use the LAD. The interviews lasted for 30 minutes for each person and a link was provided for them to access to the LAD. Before they began, a brief explanation of the LAD was provided and the teaching assistants were put in a scenerio where they were in charge of a particular exercise group in the Summer semester of 2022.

While they used the LAD, questions were asked to drive the purposeful use of the LAD.The questions were chosen from the original needs of the teachers which were collected before the development of the LAD. The questions asked during the test of the LAD include:

- What errors are most problematic for the students in your exercise group and in general?
- Which students do you feel are struggling with the exercises solutions?
- For the struggling students, do you think their submission behaviours affect their performance in the exercises?
- Which tasks do you think have proven difficult for students to solve?
- Do you think students overall performed better this semester than in the last semester?

After the interview, the TAs were given a 30-item questionnaire. With 29 questions having a 5 Likert scale (where ‘1’ represents strongly disagree and ‘5’ strongly agree) and 1 question being open ended, to evaluate their usage LAD. The first part of the questionnaire was about the perceived usefulness of the dashboard. The other part

Category	Example	No of Qs
Usefulness	Using the dashboard will improve my job performance.	6
System Quality	My interaction with the dashboard was clear and understandable.	5
User Satisfaction	I consider the dashboard flexible to interact with?	5
Information Quality	The information in the dashboard was in a suitable format.	2
System Use	Using the dashboard is a good idea.	5
Net Benefits	The dashboard helps me come up with new ideas regarding the exercises.	6
Future improvement	What improvements would you suggest for the dashboard , what can be done better?	1

Table 4.2: Summary of Survey Questionnaire.

of the questionnaire was developed using the DeLone and McLean (DM) model of Information System(IS) success [Petter et al., 2008] . The model is popular among researchers as it provides a useful framework for measuring and understanding IS success. Finally, the TAs were asked to give recommendations they thought would improve the dashboard. The DM model has six major success dimensions and they include:

- System quality : This includes certain measures such as : ease of learning, system flexibility, ease of use, system reliability, intuitiveness and responsiveness.
- Information quality : The desirable state or form of outputs from the system. For example: accuracy, relevance, conciseness, etc.
- Service quality : Measures the perceived satisfaction from the support that system users get from IT support staff in charge of the IS. This was not used in this study.
- System use : Measures the manner and extent to which users utilize the capabilities of the system. For example: frequency of use, amount of use, appropriateness of use,etc.
- User satisfaction : This measures the perceived level of satisfaction with the use of system.
- Net benefits : Evaluates how much the system is contributing to the users' success. It may include factors such as: enhanced decision-making, increased efficiency, higher sales, and so on.

5. Implementation

This chapter describes the implementation of the research presented in this thesis. This includes how the requirements for the dashboard were gathered, the planning and development of the prototype and also how the error analysis was realised.

5.1 Project Timeline

The thesis was conducted in stints of 7 phases. The phases can be visualised in Figure 5.1.

Phase 1

The focus of this phase was gathering the needs and problems facing the teachers. This involved interviewing teachers and formulating research questions.

Phase 2

In this phase, visualization guidelines that would be used to build the dashboard, methodologies to address the research questions were determined. These were achieved by conducting research into relevant literature. It was important to have relevant knowledge as regards to the state-of-the-art techniques in dashboard design and also review several literature relating error analysis.

Phase 3

At the beginning of phase 3, 2 mock-up designsof the LAD were made to get a sense of the design direction and layout. When the TAs selected a particular design, a low-fidelity prototype was developed. Another meeting was held after the prototype development to get feedback from the teachers.

Phase 4

The feedback from the teachers was analysed and the prototype from phase 3 underwent additional changes in order to improve the usability and address the teachers' concerns. The dashboard was then incorporated into the main SqlValidator system. Just like in the previous phase, a meeting was held with the TAs to get their feedback.

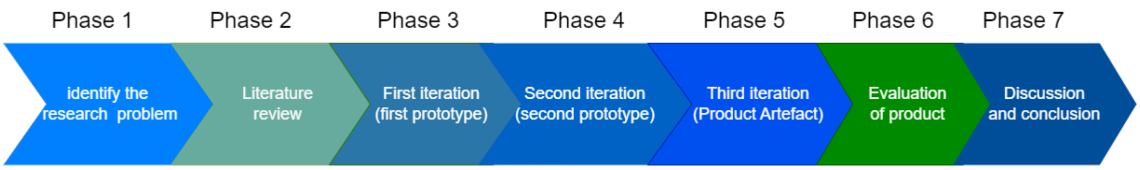

Figure 5.1: Project Timeline

Phase 5

Feedback from teaching assistants was used in the creation of this iteration of the dashboard. The dashboard underwent continuous development, testing, and incremental improvements.

Phase 6

In the phase, evaluation of the LAD was done and research questions answered. TAs utilized the dashboard to monitor the progress and errors of students.The 3 TAs were interviewed and their experience using the dashboard was gathered.

Phase 7

Summary and examination of the results obtained from the evaluation process. Future works and expectations were highlighted in this phase.

5.2 Initial Requirements

The requirements were gathered from the initial meeting with TAs as described in the 4 section.

The initial functional requirements include :

- **Data:** Query the submissions of all students and for every tasks in the selected semester.
- **Visualizations:** Implement a LAD with visualizations of error activities and represent teachers' needs with the appropriate charts.
- **User Goals:**
  - Admins will be able to understand error patterns occurring in each task.
  - An overview of the demographics of students in the course should be visible.
  - Admins should see how difficult a task is.
  - Admins should be allowed to compare visualizations with different filters.
- **Performance:** The back end and other LAD components should be as fast as possible.
- **Responsiveness:** The LAD front end should be responsive to changing screen sizes and effectively use as much screen estate as possible.

The non-functional requirements for the LAD are:

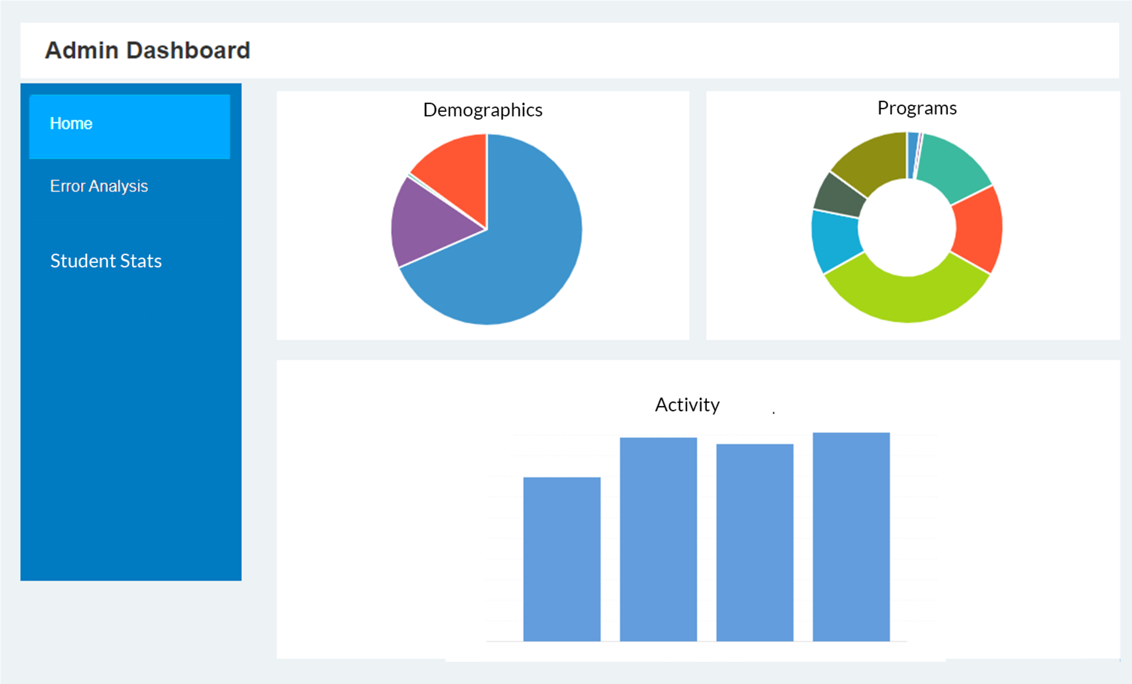

Figure 5.2: Dashboard Mock-up 1a

- **Failure States:** The LAD should handle errors gracefully.
- **User Experience (UX):** The User experience should be as pleasant as possible.
- **User Interface (UI):** The UI should be non-cluttered, visually appealing, and clean.

After a week of gathering the requirements, two mock-ups were designed in Figma and presented to the TAs inorder to get a clear design structure and direction for the LAD. The first mock-up design (Figure 5.2) separated the dashboard into logical categories with tabs enclosing related visualizations, while the second mock-up contained all visualizations on one page to provide all relevant information in a glance (Figure 5.4). The TAs selected the first mock-up which served as the background for the first Iteration.

5.3 Architecture And Used Technologies

This section describes the architecture of the LAD. Figure 5.5 shows the web architecture of LAD, it is divided into a front end, and back end which consists of a server and database. Figure 5.6 shows the acticity diagram of the LAD.

5.3.1 Front End

The front end depicts a Javascript based single page web application with multiple tabs/sections. The front end enables the user to interactively use the LAD and display relevant information in form of charts and texts. The charts were implemented using a Javascript chart library called Chart.js and asynchronous requests were made to the server using Asynchronous JavaScript And XML(AJAX).

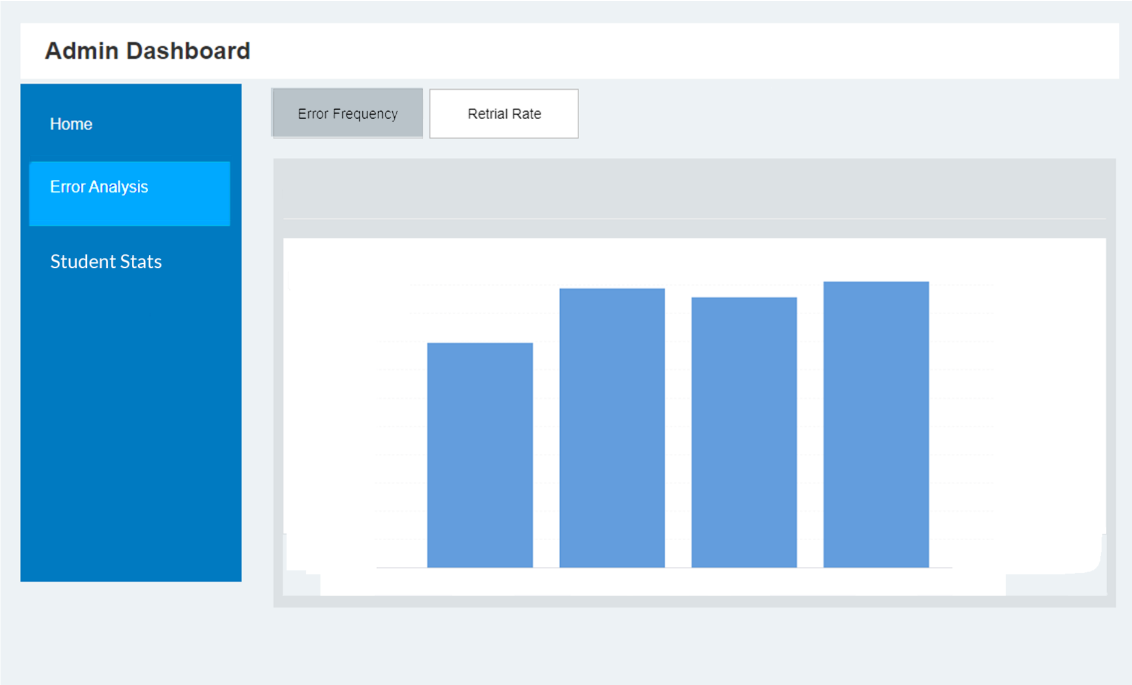

Figure 5.3: Dashboard Mock-up 1b

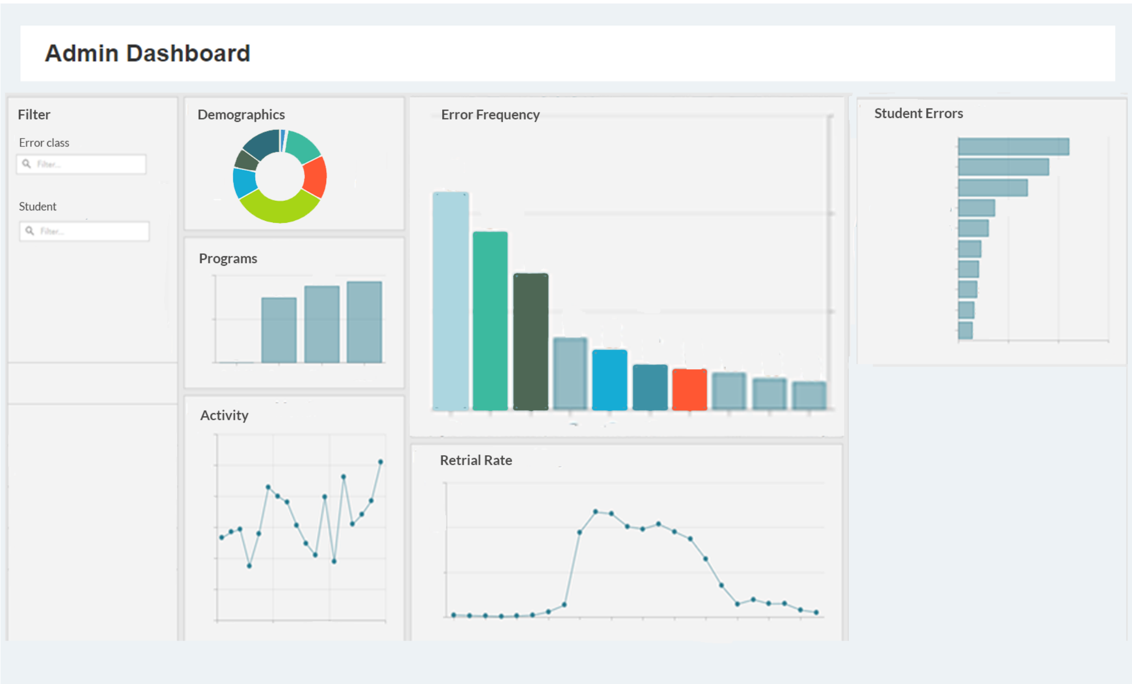

Figure 5.4: Dashboard Mock-up 2

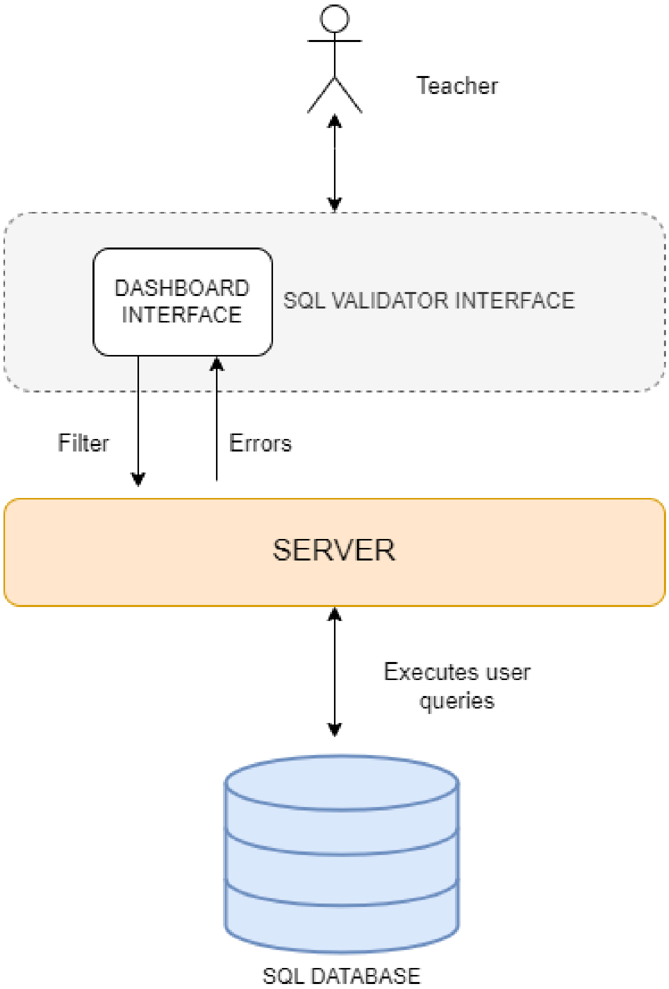

Figure 5.5: Dashboard Architecture

Chart.js

Chart.js is an open-source JavaScript library which is used for data visualization [Wikipedia contributors, 2022]. It makes it easier to display data in charts using its pre-built components. It supports 8 types of chart: line,bubble,bar, pie (doughnut),area, polar, scatter and radar chart . Chart.js utilizes the HTML5 canvas element to render its charts. It is the second most commonly used JavaScript visualization library.

5.3.2 Back End

The back end includes a web server implemented with PHP with additional connectivity to the MySQL database.

5.4 LAD Development

The development of the LAD was done in iterations, where a few changes were made in subsequent iterations. Initially, a few elements were added to the LAD, then after series of meetings with the TAs,more were added in the next iteration and so on.

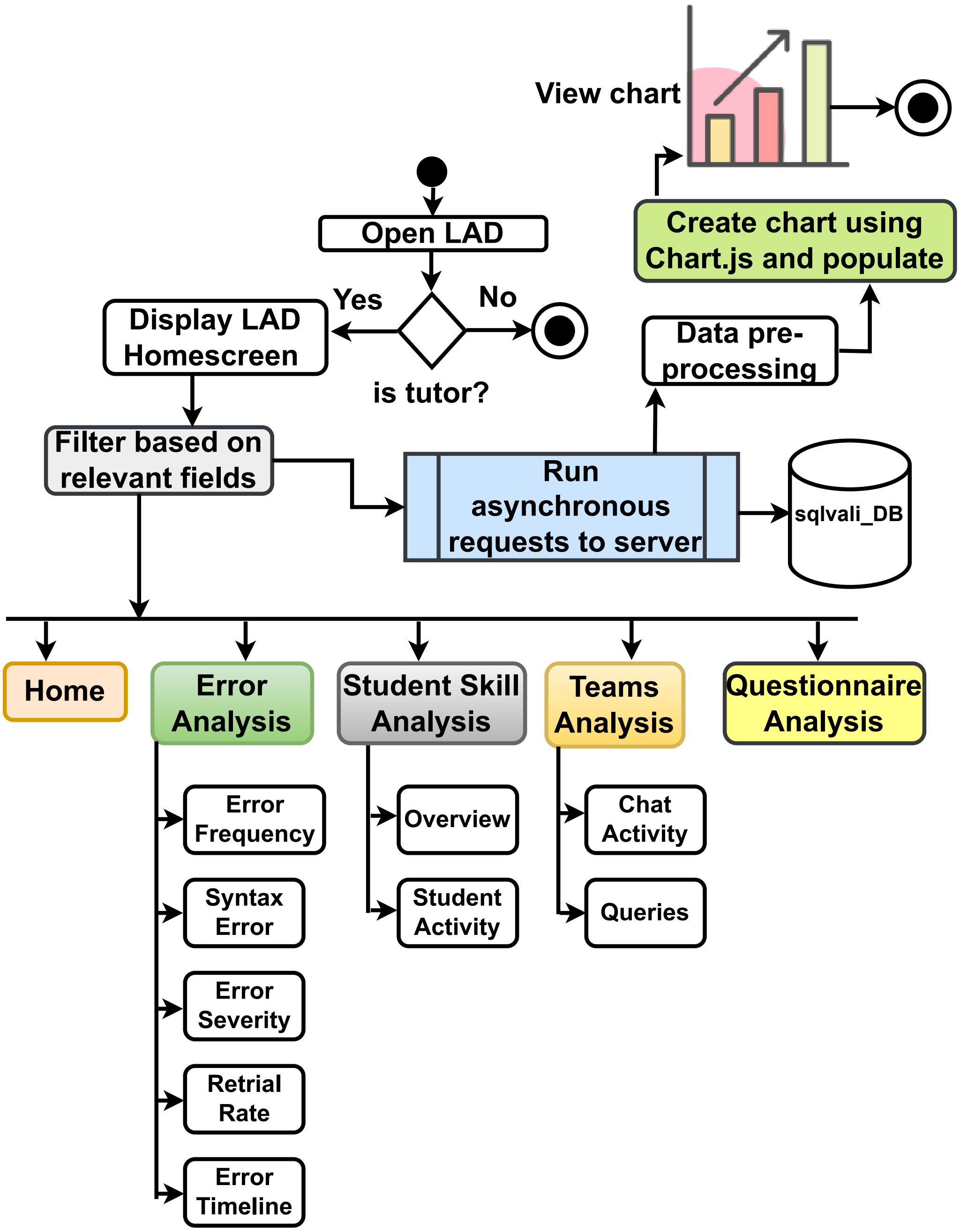

Figure 5.6: Activity diagram of the LAD

5.4.1 First Iteration

This iteration involved setting up the LAD with a local instance of the SqlValidator as I did not have administrative privileges in the SqlValidator system. The chosen mock-up design was used as a blueprint to develop the LAD. The following elements were added to the LAD:

Navigation:

The LAD was split into different screens which can be accessed depending on the focus of the user. At this stage,there is a home screen,error analysis screen and glossary are separated by tabs.

Home Screen:

The home screen is the first screen a user sees when the LAD is opened. It shows the overview of the students in the selected semester. Details components added to the home screen as can be seen in Figure 5.7 include :

- A dropdown menu shows the available semesters and allows the user to select the semester of choice.
- A text display which shows the last time the dashboard was refreshed allows the user know how current the data being visualized is.
- A pie chart which shows the demographics of students in that semester.
- A doughnut chart showing the distribution of students according to their program. This is needed since students from different programs offer the DBC course.
- A bar chart which shows a picture of the activities of the students week by week.

Error Analysis Screen:

The Error analysis screen in this iteration is divided into the error frequency screen and the retrial rate screen.

- The error frequency screen contains a bar chart which displays the frequency of the error classes for all or selected tasks. The result can be filtered by semester, task, student proficiency (as earlier filled by each student in a survey) exercise group and student. There is also an option to compare two visualizations using different filter parameters.
- The retrial rate employs a bar chart to show how many wrong submissions students make before arriving at the right answer.

Glossary:

The glossary gives more description to the error classes as they are represented by numbers in the LAD.

At the end of the first iteration, a second meeting was held with the TAs to gain feedback on the LAD. The meeting was held via Zoom.

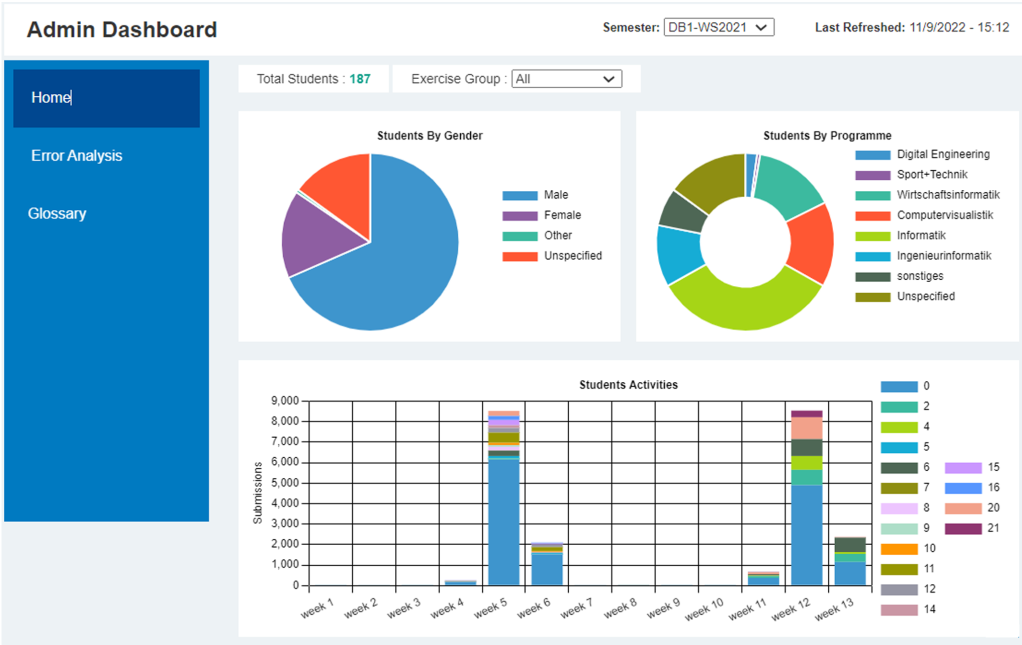

Figure 5.7: First iteration of LAD - Home Screen

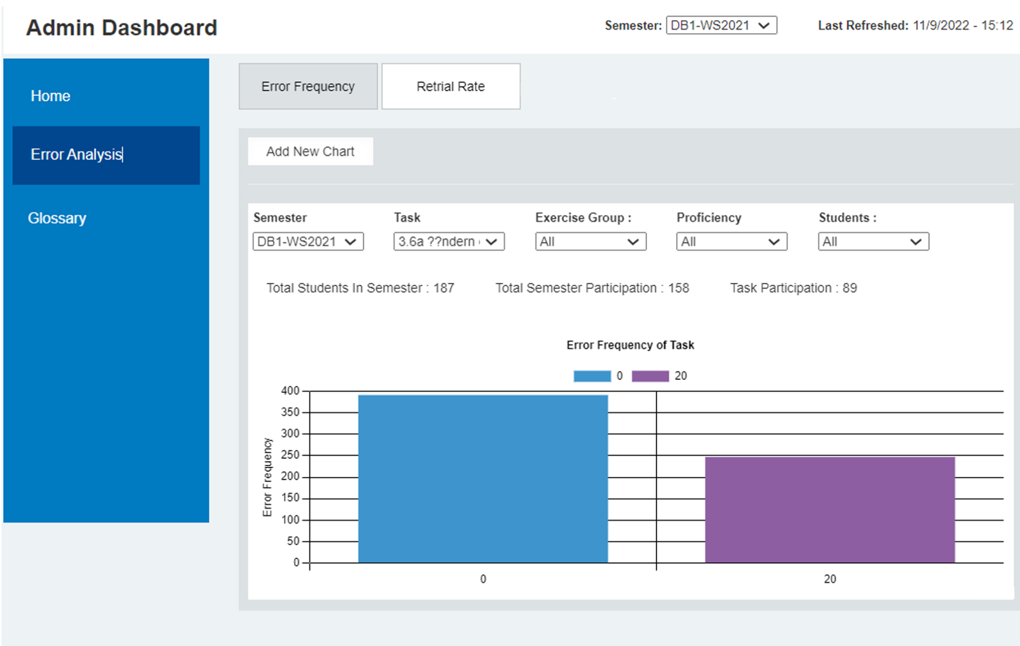

Figure 5.8: First iteration of LAD - Error Analysis Screen - Error Frequency

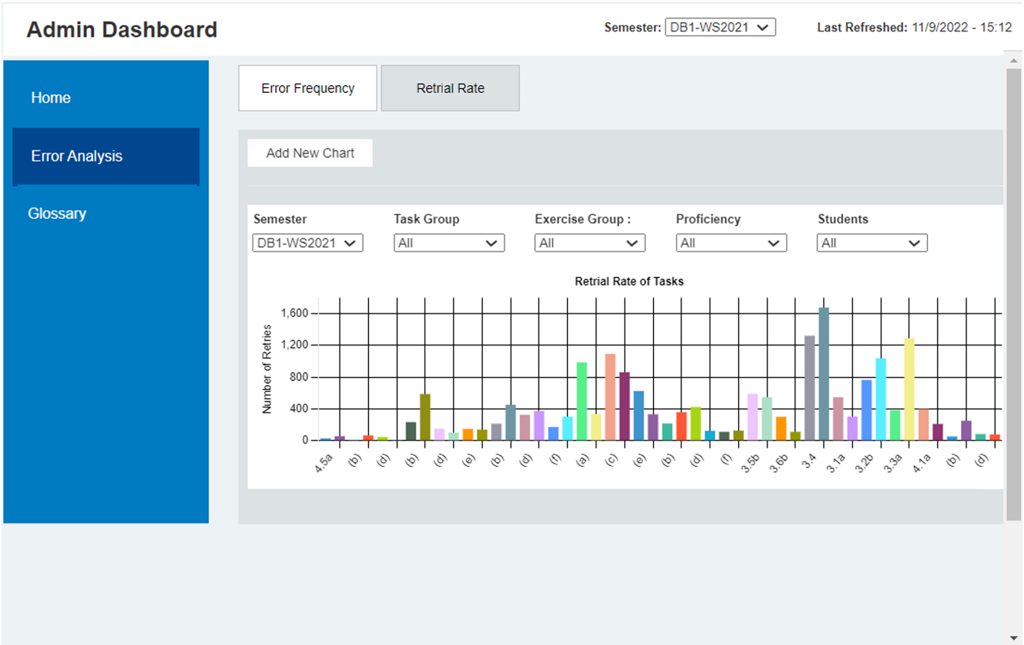

Figure 5.9: First iteration of LAD - Error Analysis Screen - Retrial Rate

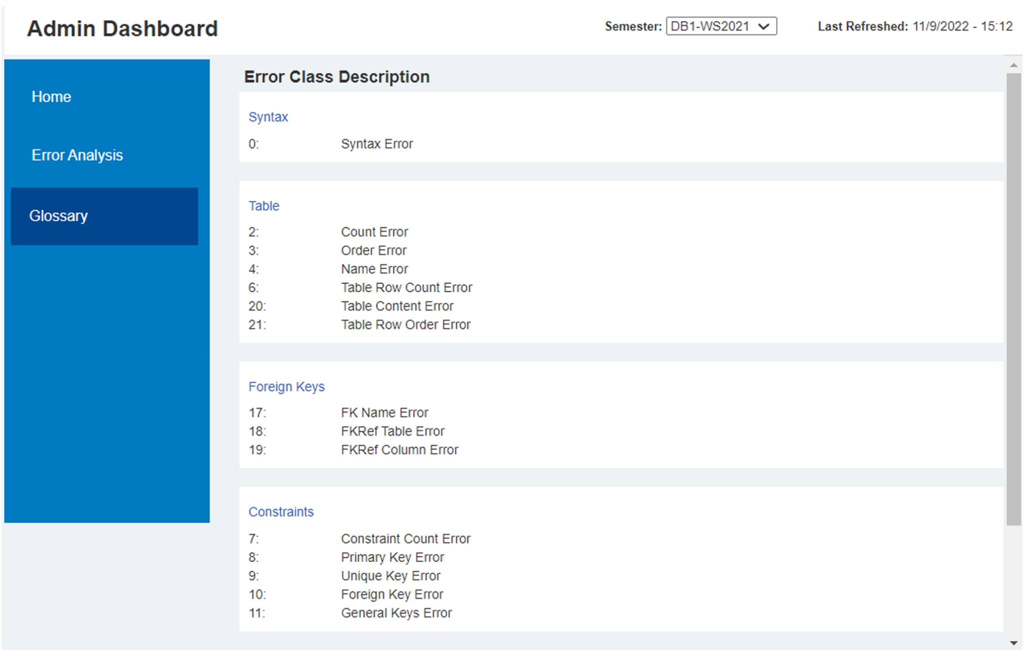

Figure 5.10: First iteration of LAD - Glossary Screen

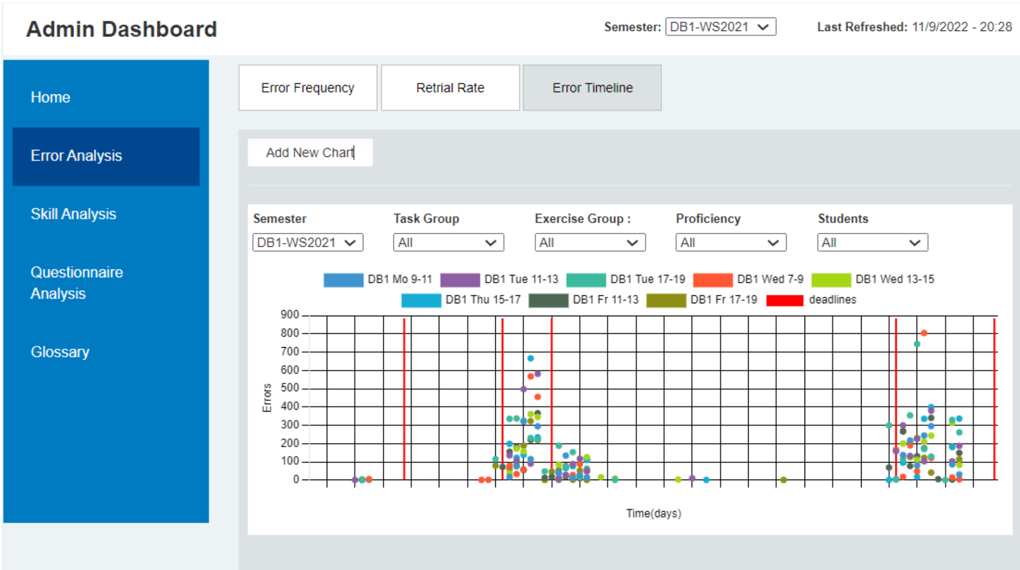

Figure 5.11: Second iteration of LAD - Error Timeline

5.4.2 Second Iteration

More features were suggested during the meeting with the teaching assistants. The changes include :

- The TAs wanted to be able to view the amount of errors and submissions with respect to the deadline.
- Since the students filled a questionnaire at the start of the semester, the TAs needed a new screen in the LAD to visualize students' answers to the questions. This was achieved by creating the **questionnaire analysis** screen which contains a chart which could be changed from one type to another to give different perspective of the data. The chart could be changed to any of the following types : stacked bar, box plot, radar and line charts.
- The TAs desired a separate screen to monitor students' submissions and trend of their errors throughout the course of the semester. This screen was called the **skill analysis** as it would help see students who are really improving.

The new requirements were incorporated into the dashboard. At the end of the iteration, another meeting was held with the teaching assistants.

5.4.3 Third Iteration

The feedback gotten from the meeting subsequent to the second iteration led to new features being added to the LAD. The features include :

- A print function was added to charts to enable exporting.
- The TAs desired to see which errors were difficult for students to fix which led to the addition of the error severity tab.

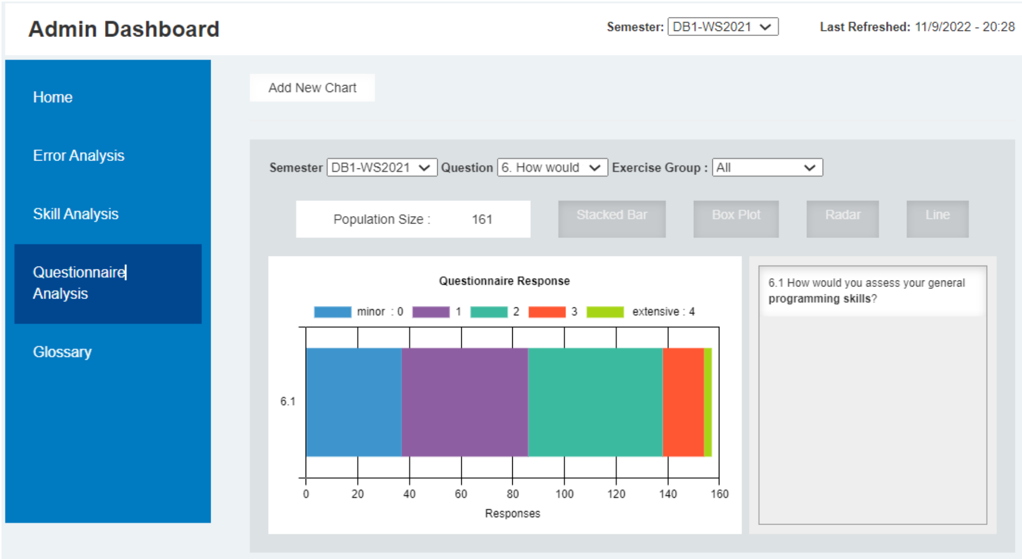

Figure 5.12: Second iteration of LAD - Questionnaire Analysis

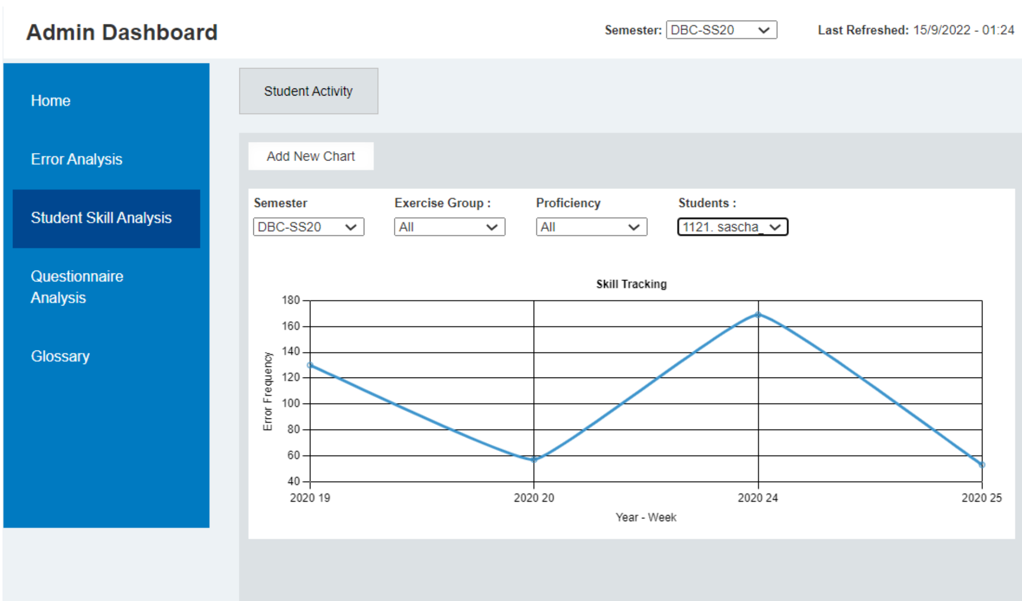

Figure 5.13: Second iteration of LAD - Skill Analysis

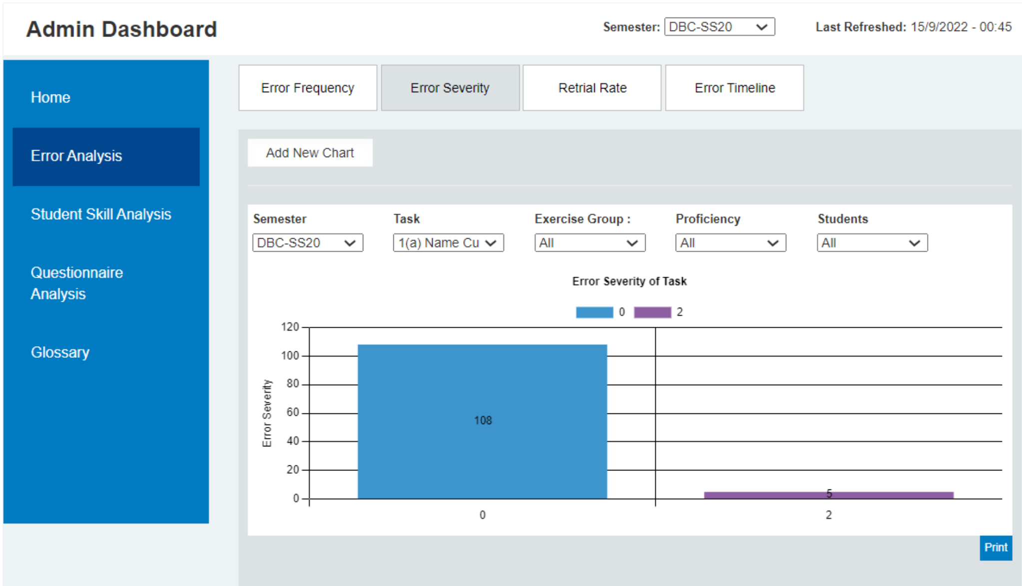

Figure 5.14: Third iteration of LAD - Error Severity

- A feature to identify struggling students was added. This showed the time students actively spent on the task and how much errors they had.This uses a scatter plot with the y-axis as number of errors, and x-axis as the time-on-task and each point representing a student.
- To track students behaviours when it came to procrastination, scatter plots were made to view students start gap, end gap and effort density relative to the amount of errors made in the task. This can be seen in Figure 5.16

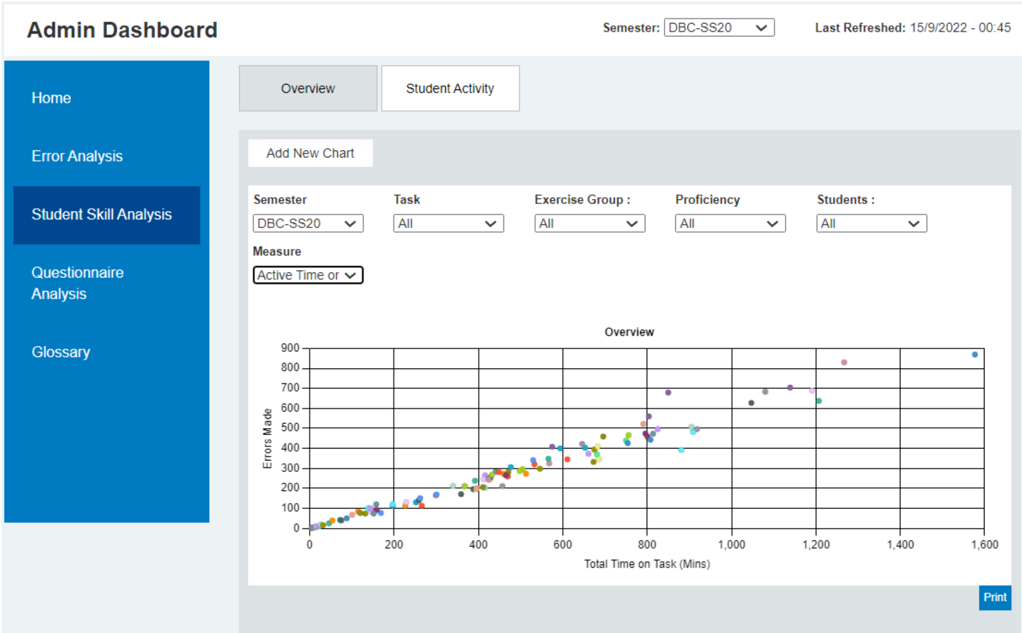

Figure 5.15: Third iteration of LAD - Time On Task

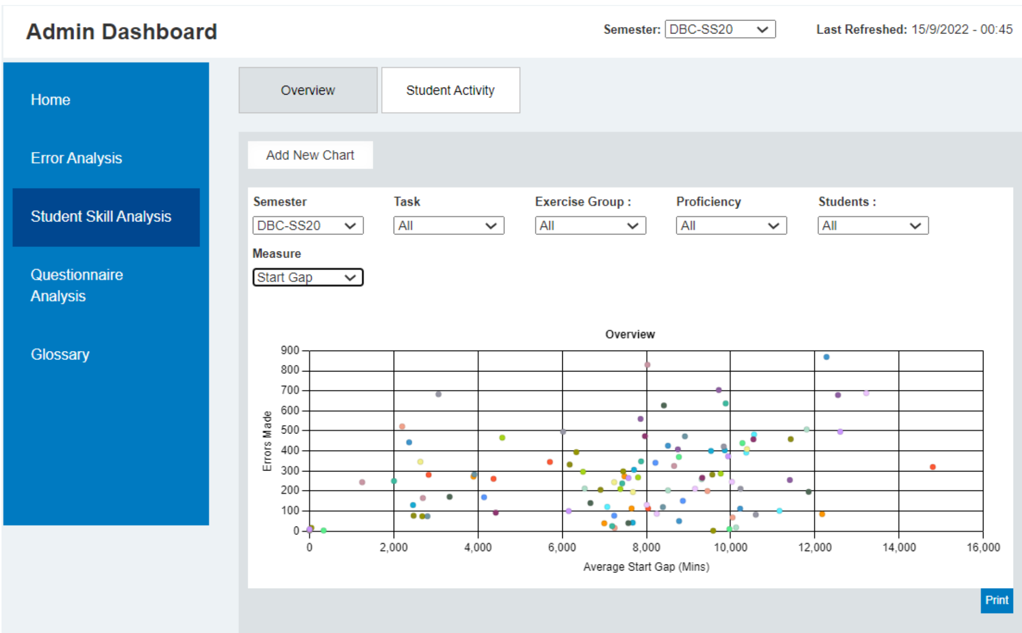

Figure 5.16: Third iteration of LAD - Start Gap

6. Evaluation And Results

6.1 Results

As mentioned previously, the two semesters being considered in this work are SS2020 and WS2021. Ss2020 recorded a total of 22,702 submissions from 104 students while ws2021 recorded a total of 22,520 submissions from 158 students.

A total of 21 error classes were derived from the SqlValidator as seen in Table 2.2.

6.1.1 User Testing

The TAs did tasks on the LAD with several prompts given to them and also filled a questionnaire at the end which measures the LAD's usefulness among other parameters. Table 6.1 shows the prompts given to the TAs and the observations from how they performed the subsequent tasks.

The questionnaire followed the DM model of Information System(IS) success and had a five point Likert scale with 1 signifying strongly disagree and 5 signifying strongly agree and one open ended question to get feedback on what could be improved. Table 6.2 summarizes the results.

Usefulness :

The TAs involved in the test were able to perform useful tasks which were related to the requirements of the LAD initially collected. Also, the questionnaire answers to usefulness related questions were high (Total Mean : 4.27, SD : 0.37).

System Quality :

The LAD has many sections which while intuitive, takes time to fully learn and master. This is observed by the answer to question "It will be easy for me to master the use of the dashboard" and "Learning to operate the dashboard was easy for me.", both which received a moderate response (Mean : 3.33, SD : 0.47). Overall, the TAs perceived the the system quality of the LAD to be moderately high (Mean : 3.86, SD : 0.56).

Prompt	**Observation**
What errors are most problematic for the students in your exercise group and in general?	The TAs opened the Error Analysis to check for most frequent errors. One TA additionally explored the Error Severity tab to make a final conclusion while others needed extra hints to do same.
Which students do you feel are struggling with the exercises solutions?	All TAs explored the Student Skill Analysis to identify students who spent more time on exercises and at the same time had more errors.
For the struggling students, do you think their submission behaviours affect their performance in the exercises?	Given that the TAs could identify the struggling students from the previous prompt, they tried to answer the prompt by exploring more measures such as Start gap, End gap, etc. It was observed that all three TAs needed reminders of what the measures truly signified.
Which tasks do you think have proven difficult for students to solve?	The TAs could identify the tasks with the most retrial rates per student as the tasks students seemed to find difficult.
Do you think students overall performed better this semester than in the last semester?	The TAs tried to compare the charts from different semesters by using the semester drop down and also by adding more charts to view multiple semesters simultaneously.

Table 6.1: Prompts and observations while the TAs used the LAD

User Satisfaction :

The TAs perceived the LAD to be flexible, easy is understand and interact with. Overall, the TAs were satisfied with the LAD (Mean : 3.93, SD : 0.70).

Information Quality :

The information quality was rated to be moderately high by the TAs (Mean : 3.50, SD : 0.47).

System Use :

Given that the LAD is built ontop of the SQLValidator, it presented a sense of continuity and familiarity to the TAs. The TAs thought the dashboard was a good idea and would recommend it to their peers. (Overall Mean : 3.95, SD : 0.74).

Net Benefits :

The TAs found found that the LAD could be beneficial to their everyday work. Benefits such as aiding decision making, improving understanding of students behaviours,etc were noted. The net benefit of the LAD was perceived as high with Overall Mean : 4.00 and SD : 0.55.

Future Improvement :

The TAs answered an open ended question on what improvements they thought the LAD needed. Responses included : ”I will like to quickly compare students' performance per semester in terms of percentage increase or decrease of error rate”, ”students should be able to also self reflect with the dashboard not just teachers”.

Category	Items	Mean	SD
Usefulness	1. Using the dashboard helped me understand students' exercise engagements more quickly.	4.33	0.47
	2. Using the dashboard will improve my job performance.	4.00	0.82
	3. The use of the dashboard will increase my productivity.	4.00	0
	4. Using the dashboard will enhance my effectiveness in my job.	3.67	0.47
	5. Using the system will make my job easier.	4.67	0.47
	6. I found the dashboard to be useful.	5.00	0
	Sub total	4.27	0.37
System Quality	7. Learning to operate the dashboard was easy for me.	3.33	0.47
	8. Learning to operate the dashboard was easy for me.	4.33	0.94
	9. My interaction with the dashboard was clear and understandable	4.33	0.94

	10. It will be easy for me to master the use of the dashboard.	4.00	0
	11. It will be easy for me to master the use of the dashboard.	3.33	0.47
	Sub total	3.86	0.56
User Satisfaction	12. I consider the dashboard flexible to interact with.	3.67	0.47
	13. The information provided by the LAD is relevant.	4.00	0.82
	14. I feel the LAD is not difficult to understand.	3.67	0.47
	15. How do you rank delays experienced while using the dashboard?	4.00	0.82
	16. How satisfied are you with the dashboard?	4.33	0.94
	Sub total	3.93	0.70
Information Quality	17. The dashboard presents information in a comprehensive way.	3.67	0.47
	18. The information in the dashboard was in a suitable format.	3.33	0.47
	Sub total	3.50	0.47
System Use	19. The dashboard is compatible with other systems I use.	4.33	0.95
	20. I will like to keep using this dashboard moving forward.	4.00	0.82
	21. I can recommend the dashboard to my colleagues.	3.67	0.94
	22. Using the dashboard is a good idea.	4.33	0.47
	23. I don't get bored quickly when using the dashboard.	3.33	0.47
	Sub total	3.95	0.74
Net Benefits	24. The dashboard helps me in decision making regarding the course, exercises and students.	4.33	0.47
	25. The dashboard helped me save time.	4.33	0.94
	26. The dashboard helps me come up with new ideas regarding the exercises.	2.67	0.47
	27. The dashboard helped me discover things which would not have been discovered otherwise.	4.67	0.47
	28. Regular use of the dashboard by teachers will positively affect the performance of students.	3.67	0.47
	29. I could understand other student behaviors beyond errors.	4.33	0.47
	Sub total	4.00	0.55

Table 6.2: Results from survey questionnaire.

6.1.2 RQ 1 How useful is the LAD in making sense of students' engagement with exercises

To answer the question of how useful the LAD is in understanding students' engagements, the responses of the three TAs who tested the LAD were gathered. The first part of the questionnaire focused on measuring how well the dashboard in meeting the needs of the teachers.

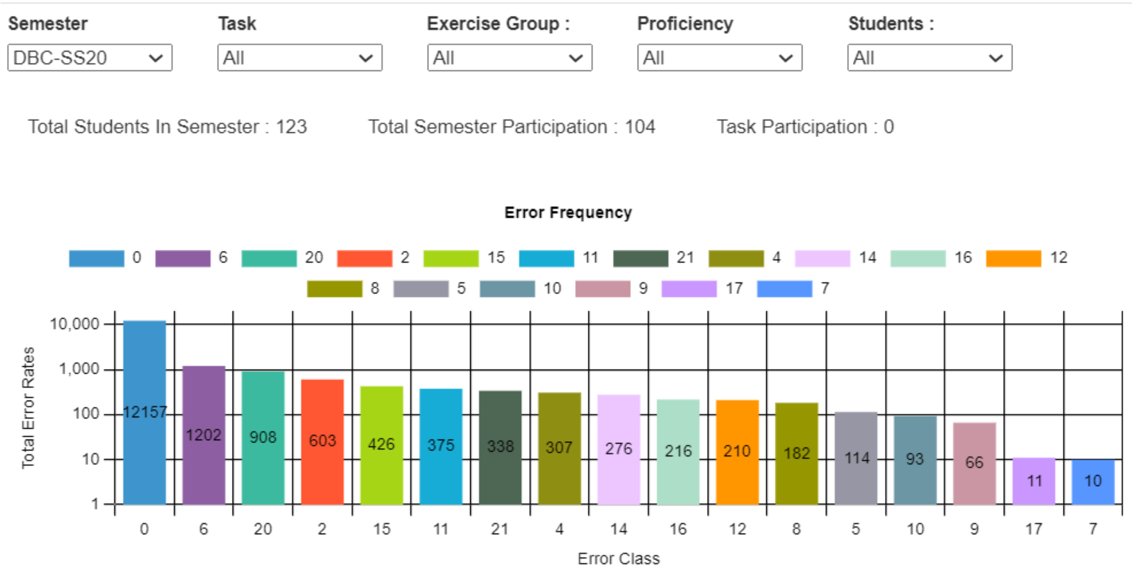

Figure 6.1: Error Frequency Chart for All Tasks in SS2020

The results showed a high level of overall usefulness (Mean=4.27, SD=0.37). This implies that the dashboard met the needs which the TAs expressed during the design phase of the LAD.

6.1.3 RQ 2 What are the common types of programming errors among students learning SQL?

The frequency distribution of SQL programming errors for each of the semesters in consideration were generated. There was a total of 17494 errors recorded in ss2020 and 15269 errors in ws2021 for a total of 42 tasks in each semester. Figure 6.3 and Figure 6.3 show the LAD visualisation for the error frequency for all tasks in the ss2020 and ws2021 semesters respectively. The Y-axis is represented in logarithmic scale to accommodate the wide range of values. The most frequent error observed from the LAD is error class "0" (Syntax error) both from ws2021 and ss2020. Error class "0" accounted for approximately 69.4% and 66.3% of all errors in the ss2020 an ws2021 semesters respectively. This results align with the results from [Poulsen et al., 2020].

For the ss2020 semester, the 10 most frequent errors are shown in Table 6.3 and a complete list can be found in Appendix A.1. Asides error class "0" which has been highlighted above others include : Table row count error (6) - 6.8%, table content error (20) - 5.2%, count error (2) - 3.4%, isDefault error (15) - 2.4%, general key error (11) - 2.1%, table Row order error (21) - 1.9%, name error (4) - 1.7%, table name error (16) - 1.2% and data type error (12) - 1.2%. This trend is very similar to the ws2021 semester except for a few minor differences (Table 6.4).

The LAD provides options to help filter the data with respect to individual tasks, exercise groups, students' proficiency and student ID. The results shows syntax error as the most common error.

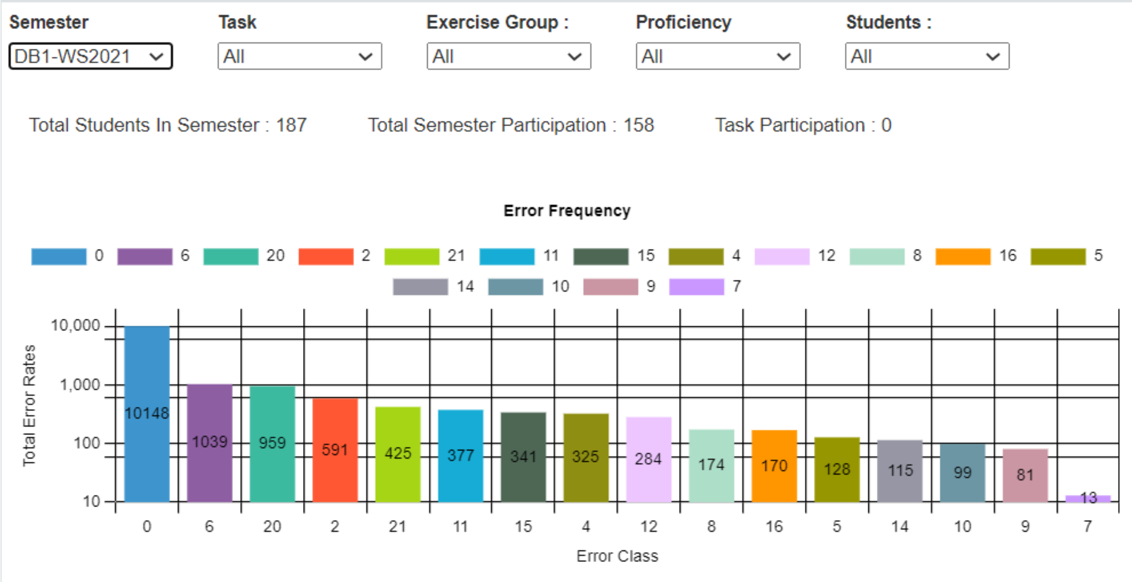

Figure 6.2: Error Frequency Chart for All Tasks in WS2021

Error Class	Frequency	%
0	12157	69.4
6	1202	6.8
20	908	5.2
2	603	3.4
15	426	2.4
11	375	2.1
21	338	1.9
4	307	1.7
16	216	1.2
12	210	1.2

Table 6.3: Frequency of Top 10 Error Classes For All Tasks in SS2020

Error Class	Frequency	%
0	10148	66.3
6	1039	6.8
20	959	6.2
2	591	3.8
21	425	2.7
11	377	2.4
15	341	2.2
4	325	2.1
12	284	1.8
8	174	1.1

Table 6.4: Frequency of Top 10 Error Classes For All Tasks in ws2021

Error Class	Percent of Failed Final submissions
0	49 (74.2%)
6	8 (12.1%)
2	7 (10.6%)
4	2 (3.0%)

Table 6.5: Last Error Class Encountered Before Student Gives Up On The Problem in SS2020

Error Class	Percent of Failed Final submissions
0	138 (85.7%)
20	8 (4.9%)
6	8 (4.9%)
2	6 (3.7%)
11	1 (0.6%)

Table 6.6: Last Error Class Encountered Before Student Gives Up On The Problem in wS2020

6.1.4 RQ 3 What SQL errors classes are the most severe for students

As described in Chapter 4, the error severity is the product of the (average) error duration and error frequency. Since the duration of errors considers the number of attempts it takes to resolve an error, special notice was taken to identify errors which could not be resolved by the students.

Resolution Status

The SQLValidator records tasks success which shows whether the last solution by a student was submitted with or without errors. With this information, errors which occurred in the last submissions were classified as unresolved.

Table 6.5 and Table 6.5 show the last errors students encountered prior to quitting the tasks. In ss2020 and ws2021 semesters respectively, error class 0 occurs 74.2% and 85.7%, error class 6 occurs 12.1% and 4.9% and error class 2 occurs 10.6% and 3.7%. Error class 4 occurs 3% of the time in ss2020, and error class 20 occurs 4.9% and error class 11 occurs 0.6% in ws2021.

In ss2020 4 error classes were seen to be left unresolved by students while 5 errors classes recorded unresolved attempts in ws2021.

The LAD provides options to help filter the severity data with respect to individual tasks, exercise groups, students' proficiency and student Figure A.1.

Table 6.7 and Table 6.8 show the error severity results for the ss2020 and ws2021 semesters respectively. Error class "0" was seen to be the most severe error encountered by students.

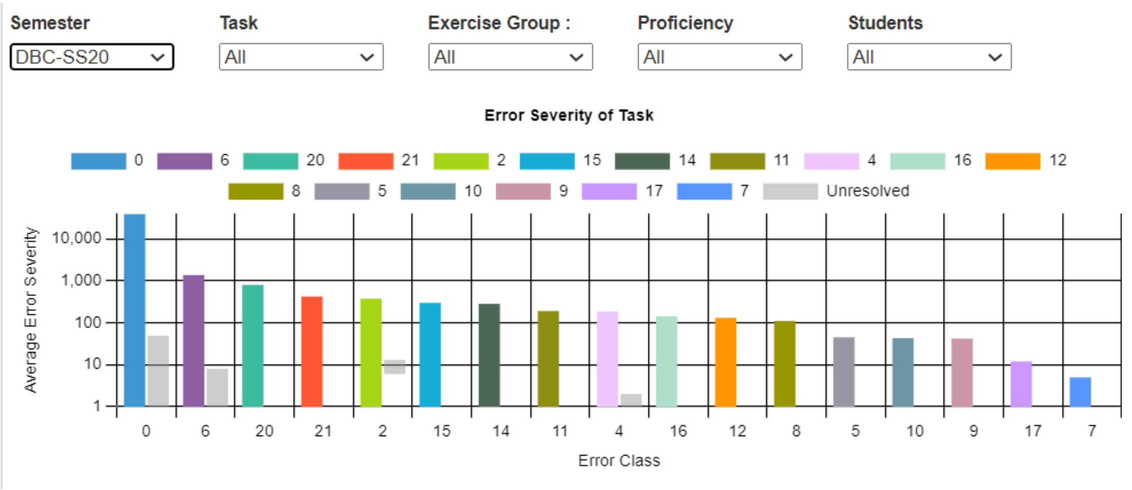

Figure 6.3: Error Severity Chart for All Tasks in sS2020

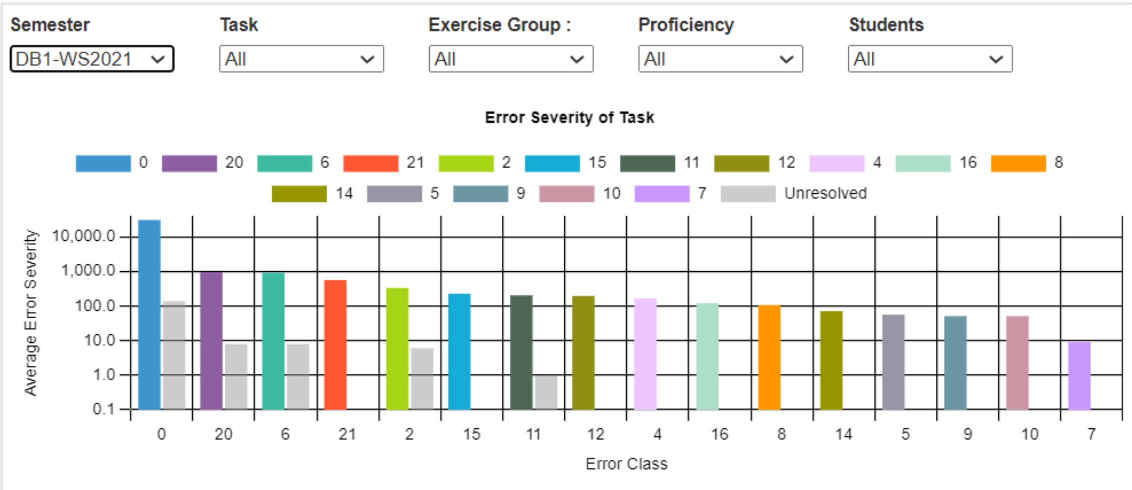

Figure 6.4: Error Severity Chart for All Tasks in wS2021

Error Class	Severity	Total	Resolved	Unresolved	Average Duration (attempts to fix)
0	38541	12157	12108 (99.5%)	49 (0.5%)	4
6	1365	1202	1194 (99.3%)	8 (0.7%)	2
20	818	908	908(100%)	0	1
21	443	338	338(100%)	0	2
2	375	603	596 (98.8%)	7(1.2%)	1
15	298	426	426 (100%)	0	1
14	282	276	276(100%)	0	2
11	190	375	375(100%)	0	1
4	186	307	305 (99.3%)	2(0.7%)	1
16	144	216	216 (100%)	0	1

Table 6.7: Error Severity and Duration: 10 Most Severe Error Classes For All Tasks in SS2020

Error Class	Severity	Total	Resolved	Unresolved	Average Duration (attempts to fix)
0	30593	10148	10010 (98.6%)	138 (1.4%)	4
20	955	959	951(99.1%)	8 (0.9%)	2
6	918	1039	1031 (99.2%)	8 (0.8%)	1
21	562	425	338(100%)	0	2
2	333	591	585 (98.9%)	6(1.1%)	1
15	227	341	426 (100%)	0	1
11	203	377	376(99.7%)	1 (0.3%)	1
12	195	284	284(100%)	0	1
4	165	325	325 (100%)	0	1
16	120	170	170 (100%)	0	1

Table 6.8: Error Severity and Duration: 10 Most Severe Error Classes For All Tasks in WS2021

Measure	**Average Error Per task**	
	SS2020	**WS2021**
Start Gap	-0.09	-0.08
End/Finish Gap	-0.17	-0.13
Effort Density	-0.77	-0.58

Table 6.9: Correlation coefficients between Start Gap, End Gap, Effort density, and average error rate in SS2020 and WS2021 semesters

6.1.5 RQ 4 Is there any correlation between submission behaviour and frequency of errors committed by students learning SQL?

To answer this research question, measures that could indicate procrastination and submission behaviours were analyzed.

The results are summarized in Table 6.9 for Start gap, End gap and Effort density.

Start Gap :
There was a very weak negative correlation observed between the Start Gap and average number of errors per task. In SS2020, the correlation coefficient is -0.09, and -0.08 for the WS2021 semester (Figure 6.5).

End/Finish Gap :
There was a weak negative correlation observed between the End/Finish Gap and average number of errors per task. In SS2020, the correlation coefficient is -0.17, and -0.13 for the WS2021 semester.

Effort Density:
There was a strong negative correlation observed between the Effort density and average number of errors per task. In SS2020, the correlation coefficient is -0.77, and -0.58 for the WS2021 semester .

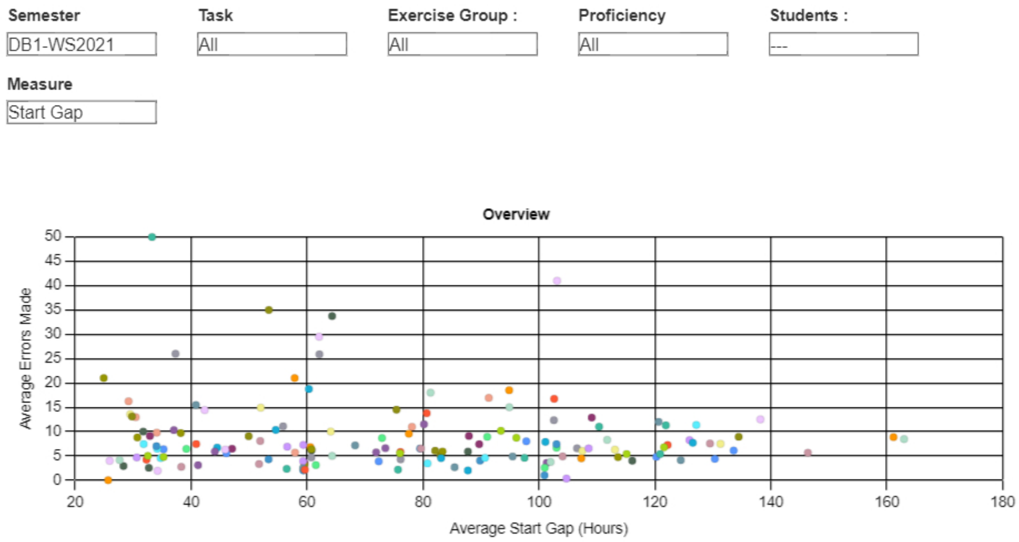

Figure 6.5: Scatter plot showing relationship between Start Gap and Average number of errors in WS2021 semester

6.2 Discussion

With the evaluation of the LAD, it is observed that it satisfies the requirements initially collected in the design phase. The results of the questionnaire showed a high level of usefulness (Mean : 4.27, SD : 0.37). Other measures of evaluation such as net benefits, user satisfaction, system quality, system use and information quality showed satisfactory results (Table 4.2).

With regards to the most frequent and severe errors which teachers should be wary of, error class "0" (syntax error) was observed to be both the most frequent and severe for students. The top 10 most frequent and sever errors for the two semesters can be seen in tables 6.3, 6.4, 6.7, and 6.8. The TAs alluded that this information could assist them in focusing on areas where students need help (Table 6.1).

In addition, the results showed that the behaviour of students as measured by when they started and stopped solving the assignments(Start gap and End gap), did not hugely affect the amount of errors they made, while Effort density however showed a strong relationship with students errors. This was not expected as higher effort density is usually associated with procrastination which tends to lead to poorer performances [Park et al., 2018].

Considering that a weak correlation was observed between start gap and finish gap and the average number of errors, it is not sufficient to say that the time of commencement or submission of tasks had a huge impact on students. Furthermore, our results showed that students who completed their assignments within fewer days/time recorded less errors when compared to students who completed the assignments in several days or longer time frame. This was shown by a strong negative correlation between Effort density and average number of errors (Table 6.9).

7. Conclusion

In conclusion, this thesis aimed to develop a LAD for a web-based interactive tool called SQLValidator which helps students learn and practice SQL related tasks. The iterative design-based research approach was employed and guidelines for dashboard design were followed. The data used in this thesis was obtained from the SQLValidator system and teacher needs were collected and analyzed using a Thematic analysis.

The LAD was designed to provide teaching assistants with a more comprehensive overview of course activity, reflect on the teaching processes they employ, and to identify where students need help. Through clear and relevant visual representation in the LAD, key strategies and decision making can be carried out by instructors. The LAD was also intended to help instructors track patterns and analyze students' performance in a more efficient way.

The LAD was tested by the TAs and the results showed that the LAD was useful in understanding students' exercise engagements more quickly and improving the teaching assistants' job performance. The teaching assistants also reported that the LAD would increase their productivity and enhance their effectiveness in their job. They found the dashboard to be useful and easy to operate. They also considered the dashboard to be flexible and to provide relevant information.

Overall, the results show that the LAD was well received by the teaching assistants, who found it to be a useful tool for monitoring students' progress and understanding their errors. The LAD will be implemented into the SQLValidator system and it is expected that regular use of the dashboard by teachers will positively affect the performance of students.

7.1 Limitations and Future work

The thesis was conducted with 2 TAs in the design phase and 3 TAs in the evaluation phase. It would be valuable to conduct a larger-scale evaluation of the LAD in order to gather more data on its effectiveness in improving student learning outcomes.

Additionally, it would be interesting to explore other ways where the LAD could be utilized to support other aspects of learning and teaching, such as tracking students reading, predicting students performance or enabling student self-reflection.

Another direction that would be beneficial to explore is expanding the scope of the dashboard to include other courses and subjects, to understand the generalizability of the design. Furthermore, It could be beneficial to further investigate the effect of collaborative work on how students perform in the SQL tasks.

Appendix

A.1 Evaluation Results

Error Class	Frequency	%
0	12157	69.4
6	1202	6.8
20	908	5.2
2	603	3.4
15	426	2.4
11	375	2.1
21	338	1.9
4	307	1.7
16	216	1.2
12	210	1.2

Table A.1: Frequency of Error Classes For All Tasks in SS2020

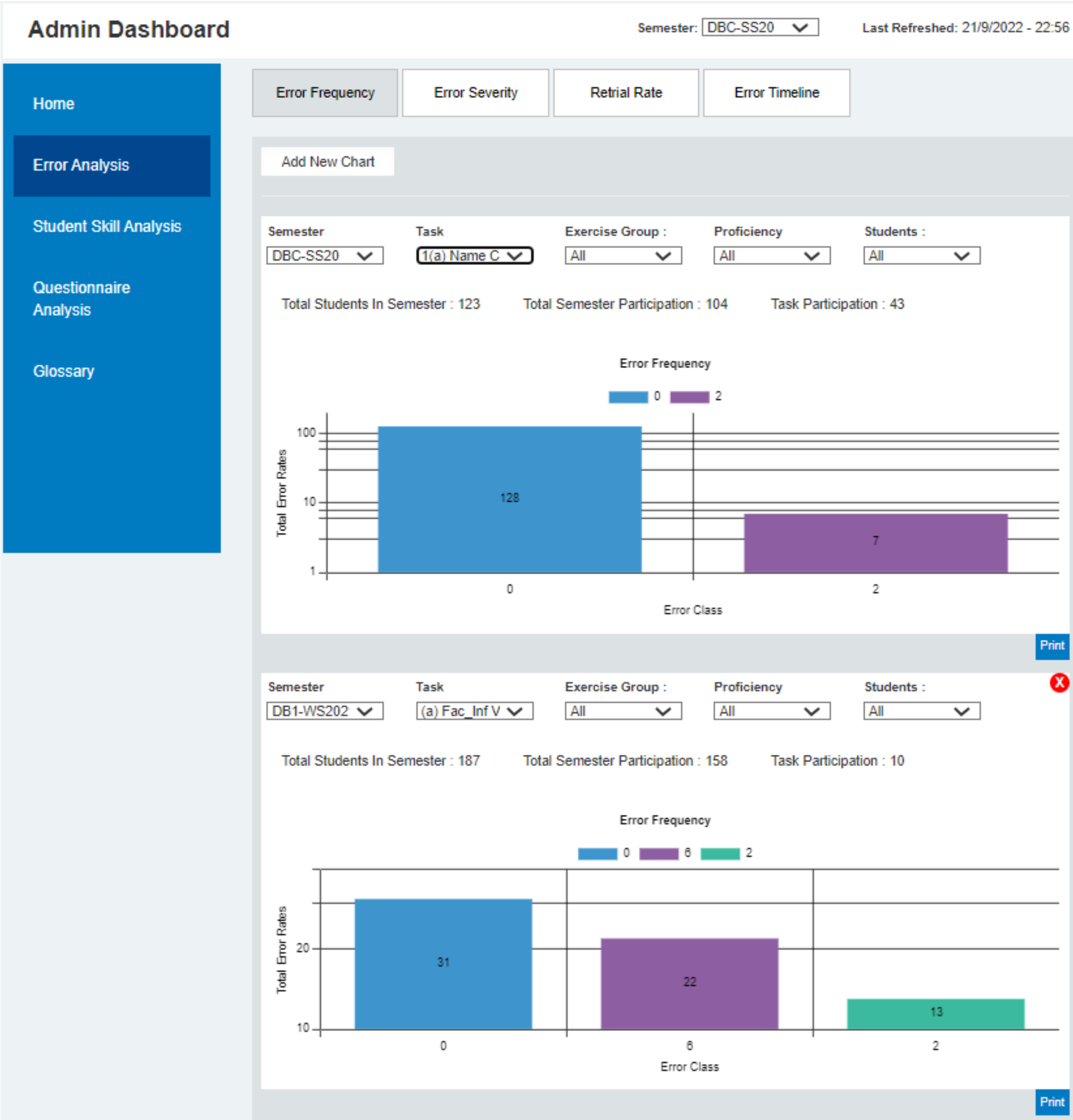

Figure A.1: Error Frequency Chart for Exercise Sheet 8 - Task 1(a) in ss2020 and WS2021 semesters

Error Class	Frequency	%
0	10148	66.3
6	1039	6.8
20	959	6.2
2	591	3.8
21	425	2.7
11	377	2.4
15	341	2.2
4	325	2.1
12	284	1.8
8	174	1.1

Table A.2: Frequency of Error Classes For All Tasks in ws2021

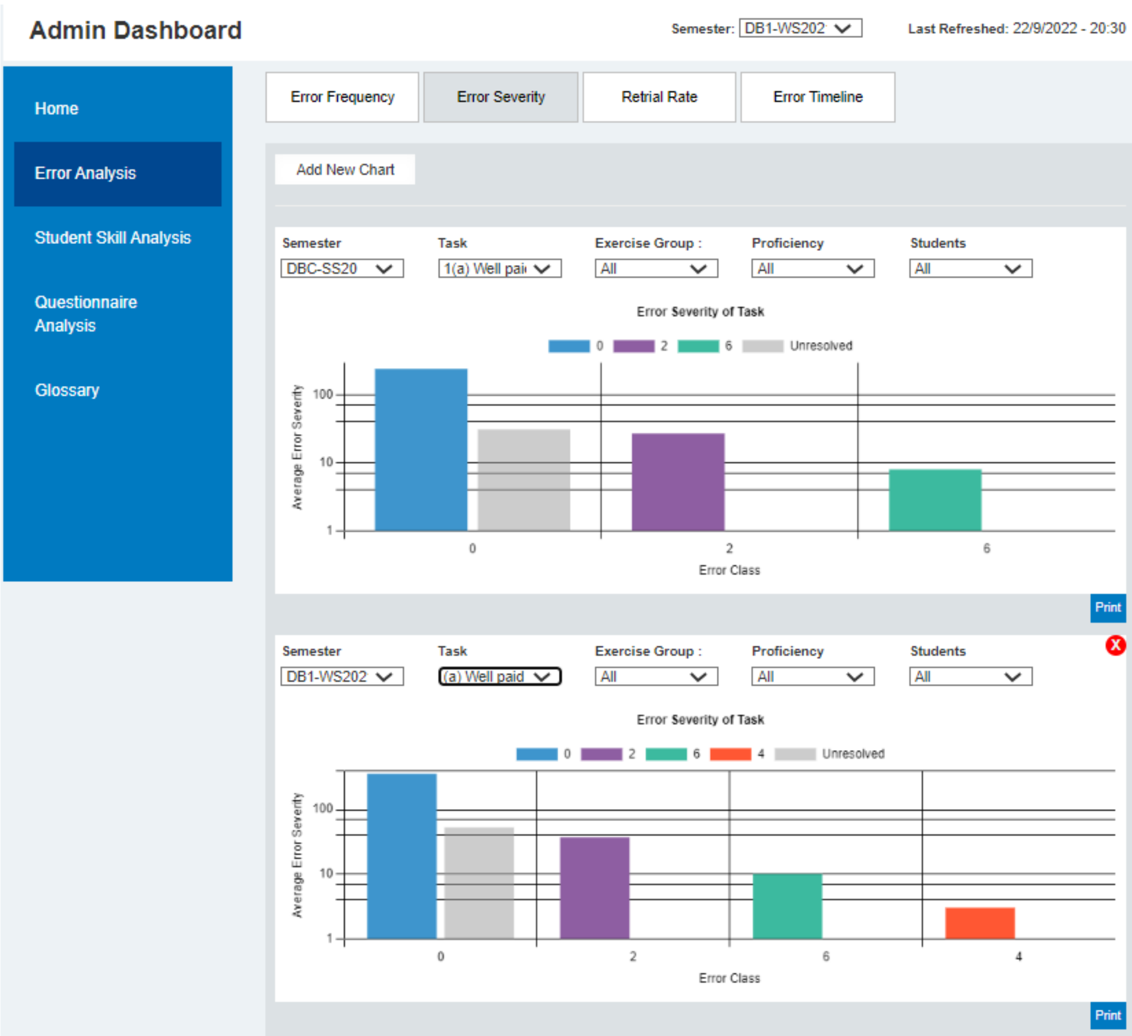

Figure A.2: Error Severity Chart for one task in ss2020 and WS2021 semesters

Error Class	Severity	Total	Resolved	Unresolved	Average Duration (attempts to fix)
0	38541	12157	12108 (99.5%)	49 (0.5%)	4
6	1365	1202	1194 (99.3%)	8 (0.7%)	2
20	818	908	908(100%)	0	1
21	443	338	338(100%)	0	2
2	375	603	596 (98.8%)	7(1.2%)	1
15	298	426	426 (100%)	0	1
14	282	276	276(100%)	0	2
11	190	375	375(100%)	0	1
4	186	307	305 (99.3%)	2(0.7%)	1
16	144	216	216 (100%)	0	1
12	131	210	210 (100%)	0	1
8	110	182	1826 (100%)	0	1
5	46	114	114 (100%)	0	1
10	45	93	93 (100%)	0	1
9	42	66	66 (100%)	0	1
17	12	11	11 (100%)	0	2
7	5	10	10 (100%)	0	1

Table A.3: Error Severity and Duration: Error Classes For All Tasks in SS2020

Error Class	Severity	Total	Resolved	Unresolved	Average Duration (attempts to fix)
0	30593	10148	10010 (98.6%)	138 (1.4%)	4
20	955	959	951(99.1%)	8 (0.9%)	2
6	918	1039	1031 (99.2%)	8 (0.8%)	1
21	562	425	338(100%)	0	2
2	333	591	585 (98.9%)	6(1.1%)	1
15	227	341	426 (100%)	0	1
11	203	377	376(99.7%)	1 (0.3%)	1
12	195	284	284(100%)	0	1
4	165	325	325 (100%)	0	1
16	120	170	170 (100%)	0	1
8	105	174	174 (100%)	0	1
14	71	115	115(100%)	0	2
5	56	128	128(100%)	0	1
9	51	81	81(100%)	0	1
10	51	99	99(100%)	0	1
7	9	13	13(100%)	0	1

Table A.4: Error Severity and Duration: Error Classes For All Tasks in WS2021

Bibliography

Javascript line charts amp; graphs, Dec 2022. URL https://canvasjs.com/html5-javascript-line-chart/. (cited on Page 9)

Alireza Ahadi, Vahid Behbood, Arto Vihavainen, Julia Prior, and Raymond Lister. Students' syntactic mistakes in writing seven different types of sql queries and its application to predicting students' success. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, pages 401–406, 2016. (cited on Page 19)

June Ahn, Fabio Campos, Maria Hays, and Daniela DiGiacomo. Designing in context: Reaching beyond usability in learning analytics dashboard design. *Journal of Learning Analytics*, 6(2):70–85, 2019. (cited on Page 13)

Samuel Soma Ajibade and Anthonia Adediran. An overview of big data visualization techniques in data mining. *International Journal of Computer Science and Information Technology Research*, 4(3):105–113, 2016. (cited on Page 7)

Ella Albrecht. Analysis of students' programming knowledge and error development. 2022. (cited on Page 25)

Amjad Altadmri and Neil CC Brown. 37 million compilations: Investigating novice programming mistakes in large-scale student data. In *Proceedings of the 46th ACM technical symposium on computer science education*, pages 522–527, 2015. (cited on Page 18)

Peter M Asaro. Transforming society by transforming technology: the science and politics of participatory design. *Accounting, Management and Information Technologies*, 10(4):257–290, 2000. (cited on Page 14)

Tarek Azzam, Stephanie Evergreen, Amy A. Germuth, and Susan J. Kistler. Data visualization and evaluation. *New Directions for Evaluation*, 2013(139):7–32, 2013. doi: https://doi.org/10.1002/ev.20065. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/ev.20065. (cited on Page 7)

Aneesha Bakharia, Linda Corrin, Paula De Barba, Gregor Kennedy, Dragan Gašević, Raoul Mulder, David Williams, Shane Dawson, and Lori Lockyer. A conceptual framework linking learning design with learning analytics. In *Proceedings of the sixth international conference on learning analytics & knowledge*, pages 329–338, 2016. (cited on Page 2)

Paulo Blikstein. Using learning analytics to assess students' behavior in open-ended programming tasks. In *Proceedings of the 1st International Conference on Learning Analytics and Knowledge*, LAK '11, page 110–116, New York, NY, USA, 2011. Association for Computing Machinery. ISBN 9781450309448. doi: 10.1145/2090116.2090132. URL https://doi.org/10.1145/2090116.2090132. (cited on Page 1)

Robert Bodily, Judy Kay, Vincent Aleven, Ioana Jivet, Dan Davis, Franceska Xhakaj, and Katrien Verbert. Open learner models and learning analytics dashboards: a systematic review. pages 41–50, 03 2018. doi: 10.1145/3170358.3170409. (cited on Page 2)

Virginia Braun and Victoria Clarke. *Thematic analysis.* American Psychological Association, 2012. (cited on Page 23)

Rex P Bringula, Geecee Maybelline A Manabat, Miguel Angelo A Tolentino, and Edmon L Torres. Predictors of errors of novice java programmers. *World Journal of Education*, 2(1):3–15, 2012. (cited on Page 10)

Armelle Brun, Geoffray Bonnin, Sylvain Castagnos, Azim Roussanaly, and Anne Boyer. Learning analytics made in france: The metal project. *The International Journal of Information and Learning Technology*, 2019. (cited on Page 15)

Simon Buckingham Shum, Rebecca Ferguson, and Roberto Martinez-Maldonado. Human-centred learning analytics. *Journal of Learning Analytics*, 6(2):1–9, 2019. (cited on Page 13)

Mohamed Amine Chatti, Anna Lea Dyckhoff, Ulrik Schroeder, and Hendrik Thüs. A reference model for learning analytics. *International Journal of Technology Enhanced Learning*, 4(5-6):318–331, 2012. doi: 10.1504/IJTEL.2012.051815. URL https://www.inderscienceonline.com/doi/abs/10.1504/IJTEL.2012.051815. (cited on Page xiii, 5, and 7)

Marisol Clark-Ibáñez and Linda Scott. Learning to teach online. *Teaching Sociology*, 36(1):34–41, 2008. ISSN 0092055X. URL http://www.jstor.org/stable/20058625. (cited on Page 1)

Linda Corrin, PG De Barba, L Lockyear, Dragan Gašević, D Williams, Shane Dawson, R Mulder, S Copeland, and A Bakharia. Completing the loop: Returning meaningful learning analytic data to teachers. 2016. (cited on Page 17)

Carrie Demmans Epp, Ryan Perez, Krystle Phirangee, Jim Hewitt, and Keeley Toope. User-centered dashboard design: Iterative design to support teacher informational needs in online learning contexts. 04 2019. doi: 10.3102/1432967. (cited on Page 18)

Mollie Dollinger and Jason M Lodge. Co-creation strategies for learning analytics. In *Proceedings of the 8th international conference on learning analytics and knowledge*, pages 97–101, 2018. (cited on Page 14)

Wayne W Eckerson. *Performance dashboards: measuring, monitoring, and managing your business.* John Wiley & Sons, 2013. (cited on Page 7, 8, and 22)

Negin Entezari, Amirali Darvishzadeh, and Thomas Stahovich. Using an instrumented learning environment to understand procrastination. pages 1–8, 10 2018. doi: 10.1109/FIE.2018.8658662. (cited on Page 10 and 13)

Mohamed Ez-Zaouia. Teacher-centered dashboards design process. In *2nd International Workshop on eXplainable Learning Analytics, Companion Proceedings of the 10th International Conference on Learning Analytics & Knowledge LAK20*, 2020. (cited on Page 18)

Gloria Milena Fernandez Nieto, Kirsty Kitto, Simon Buckingham Shum, and Roberto Martinez-Maldonado. Beyond the learning analytics dashboard: Alternative ways to communicate student data insights combining visualisation, narrative and storytelling. In *LAK22: 12th International Learning Analytics and Knowledge Conference*, pages 219–229, 2022. (cited on Page 17)

Joseph R Ferrari, Judith L Johnson, and William G McCown. *Procrastination and task avoidance: Theory, research, and treatment*. Springer Science & Business Media, 1995. (cited on Page 10)

Stephen Few. *Information dashboard design: The effective visual communication of data*. O'Reilly Media, Inc., 2006. (cited on Page 7, 8, 21, and 22)

Gerhard Fischer, Kumiyo Nakakoji, and Yunwen Ye. Metadesign: Guidelines for supporting domain experts in software development. *IEEE software*, 26(5):37–44, 2009. (cited on Page 15)

Rohan Gupta. Data visualization which graphs should i use, Nov 2019. URL https://towardsdatascience.com/data-visualization-which-graphs-should-i-use-55e214ee9cf1. (cited on Page 9)

Suhaila Abd Halim. Development of student attendance and continuous assessment performance dashboard. *International Journal of Advanced Research in Technology and Innovation*, 3(3):111–116, 2021. ISSN 2682-8324. URL https://myjms.mohe.gov.my/index.php/ijarti/article/view/16036. (cited on Page 2 and 17)

Kenneth Holstein, Bruce M McLaren, and Vincent Aleven. Co-designing a real-time classroom orchestration tool to support teacher–ai complementarity. *Journal of Learning Analytics*, 6(2), 2019. (cited on Page 15)

Matthew C Jadud. Methods and tools for exploring novice compilation behaviour. In *Proceedings of the second international workshop on Computing education research*, pages 73–84, 2006. (cited on Page 18)

L Johnson, R. Smith, H. Willis, A. Levine, and K. Haywood. The 2011 horizon report. *New Media Consortium*, 57(10):1500–1509, 2011. URL http://net.educause.edu/ir/library/pdf/HR2011.pdf. (cited on Page 1)

D.A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002. doi: 10.1109/2945.981847. (cited on Page 8)

Mohammad Khalil and Martin Ebner. Learning analytics: principles and constraints. In *EdMedia+ Innovate Learning*, pages 1789–1799. Association for the Advancement of Computing in Education (AACE), 2015. (cited on Page xiii, 5, and 6)

Joris Klerkx, Katrien Verbert, and Erik Duval. Enhancing learning with visualization techniques. In *Handbook of research on educational communications and technology*, pages 791–807. Springer, 2014. (cited on Page 7)

Qiujie Li, Yeonji Jung, and Alyssa Friend Wise. Beyond first encounters with analytics: Questions, techniques and challenges in instructors' sensemaking. In *LAK21: 11th International Learning Analytics and Knowledge Conference*, pages 344–353, 2021. (cited on Page 2)

Amna Liaqat, Benett Axtell, Cosmin Munteanu, and Carrie Demmans Epp. Contextual inquiry, participatory design, and learning analytics: An example. In *Companion Proceedings 8th International Conference on Learning Analytics & Knowledge*, pages 1–5, 2018. (cited on Page 15)

Davin Mccall. *Novice Programmer Errors-Analysis and Diagnostics*. University of Kent (United Kingdom), 2016. (cited on Page 10 and 25)

Susan McKenney and Thomas C Reeves. *Conducting educational design research*. Routledge, 2018. (cited on Page 21)

Tamara Munzner. A nested model for visualization design and validation. *IEEE transactions on visualization and computer graphics*, 15(6):921–928, 2009. (cited on Page 18)

Victor Obionwu, David Broneske, Anja Hawlitschek, Veit Köppen, and Gunter Saake. Sqlvalidator–an online student playground to learn sql. *Datenbank-Spektrum*, 21 (2):73–81, 2021. (cited on Page xiii, xv, 10, 11, 12, and 23)

Jihyun Park, Renzhe Yu, Fernando Rodriguez, Rachel Baker, Padhraic Smyth, and Mark Warschauer. Understanding student procrastination via mixture models. *International Educational Data Mining Society*, 2018. (cited on Page 10 and 50)

Yeonjeong Park and Il-Hyun Jo. Factors that affect the success of learning analytics dashboards. *Educational Technology Research and Development*, 67(6):1547–1571, 2019. (cited on Page 17)

Stacie Petter, William DeLone, and Ephraim McLean. Measuring information systems success: models, dimensions, measures, and interrelationships. *European journal of information systems*, 17(3):236–263, 2008. (cited on Page 26)

Seth Poulsen, Liia Butler, Abdussalam Alawini, and Geoffrey L Herman. Insights from student solutions to sql homework problems. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, pages 404–410, 2020. (cited on Page 2, 18, 19, and 45)

Stanislav Pozdniakov, Roberto Martinez-Maldonado, Yi-Shan Tsai, Mutlu Cukurova, Tom Bartindale, Peter Chen, Harrison Marshall, Dan Richardson, and Dragan

Gasevic. The question-driven dashboard: How can we design analytics interfaces aligned to teachers' inquiry? In *LAK22: 12th International Learning Analytics and Knowledge Conference*, pages 175–185, 2022. (cited on Page 18)

Carlos Gerardo Prieto Alvarez. *Engaging stakeholders in the learning analytics design process*. PhD thesis, 2020. (cited on Page xiii, 21, and 22)

Matthew Sadiku, Adebowale E Shadare, Sarhan M Musa, Cajetan M Akujuobi, and Roy Perry. Data visualization. *International Journal of Engineering Research And Advanced Technology (IJERAT)*, 2(12):11–16, 2016. (cited on Page 7)

Elizabeth B-N Sanders and Pieter Jan Stappers. Co-creation and the new landscapes of design. *Co-design*, 4(1):5–18, 2008. (cited on Page xiii and 14)

Juan Pablo Sarmiento and Alyssa Friend Wise. Participatory and co-design of learning analytics: An initial review of the literature. In *LAK22: 12th International Learning Analytics and Knowledge Conference*, pages 535–541, 2022. (cited on Page 14 and 15)

Simon Buckingham Shum. Transitioning education's knowledge infrastructure: Shaping design or shouting from the touchline? In *Proceedings of International Conference of the Learning Sciences, ICLS*, 2018. (cited on Page 2)

Samuel Soma. A systematic review of data visualization methods, Nov 2016. URL https://www.academia.edu/29972986/A_Systematic_Review_of_Data_Visualization_Methods. (cited on Page 9)

James C. Spohrer and Elliot Soloway. Novice mistakes: Are the folk wisdoms correct? *Commun. ACM*, 29(7):624–632, jul 1986a. ISSN 0001-0782. doi: 10.1145/6138.6145. URL https://doi.org/10.1145/6138.6145. (cited on Page 2)

James C Spohrer and Elliot Soloway. Alternatives to construct-based programming misconceptions. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 183–191, 1986b. (cited on Page 10)

Emily S Tabanao, Ma Mercedes T Rodrigo, and Matthew C Jadud. Predicting at-risk novice java programmers through the analysis of online protocols. In *Proceedings of the seventh international workshop on Computing education research*, pages 85–92, 2011. (cited on Page 18)

Toni Taipalus and Piia Perälä. What to expect and what to focus on in sql query teaching. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, pages 198–203, 2019. (cited on Page 19)

Ashok Kumar Veerasamy, Daryl D'Souza, and Mikko-Jussi Laakso. Identifying novice student programming misconceptions and errors from summative assessments. *Journal of Educational Technology Systems*, 45(1):50–73, 2016. (cited on Page 10)

Katrien Verbert, Erik Duval, Joris Klerkx, Sten Govaerts, and José Luis Santos. Learning analytics dashboard applications. *American Behavioral Scientist*, 57(10): 1500–1509, 2013. doi: 10.1177/0002764213479363. URL https://doi.org/10.1177/0002764213479363. (cited on Page 2)

Feng Wang and Michael J Hannafin. Design-based research and technology-enhanced learning environments. *Educational technology research and development*, 53(4): 5–23, 2005. (cited on Page 21)

Wikipedia contributors. Chart.js — Wikipedia, the free encyclopedia, 2022. URL https://en.wikipedia.org/w/index.php?title=Chart.js&oldid=1128725169. [Online; accessed 12-January-2023]. (cited on Page 31)

Françeska Xhakaj, Vincent Aleven, and Bruce M McLaren. How teachers use data to help students learn: Contextual inquiry for the design of a dashboard. In *European Conference on Technology Enhanced Learning*, pages 340–354. Springer, 2016. (cited on Page 2)

I am making a statement that the thesis I am submitting was created by me and has not been previously submitted for grading or used as academic work. Additionally, I have only used the sources that are listed and any quotes or paraphrased information from those sources have been properly cited.

Magdeburg, January 9, 2023