Efficient Evaluation of CAD Skill in Individual Users

by

P. A. Di Palma

A thesis submitted in conformity with the requirements
for the degree of Bachelor of Applied Science

Department of Mechanical and Industrial Engineering

University of Toronto

Supervisor: Dr. Alison Olechowski

Efficient Evaluation of CAD Skill in Individual Users

P.A.Di Palma

Bachelor of Applied Science

Department of Mechanical and Industrial Engineering
University of Toronto

2023

Abstract

Computer-Aided Design (CAD) software is a cornerstone of modern engineering design. Efficacy in CAD is dependent on the user's skill level. A variety of methods exist to evaluate CAD skill, however there is no consensus on what protocol is the best, or what skills are "good".

In this paper, a framework is proposed as a tool to efficiently evaluate a user's skill level in CAD. The framework is developed to reveal knowledge of a user's skill through their actions and model features. The framework was applied to a design activity presented to a group of first-year engineering undergraduate students of varying skill levels.

The framework presented in this paper leverages backend analytical data and automated processes to provide a standardized protocol for evaluating the CAD skill of a large dataset of users. As an evolution to existing processes, the framework developed significantly reduces the time and effort required to evaluate CAD models.

Acknowledgments

I would like to express my sincere gratitude to Professor Alison Olechowski for her invaluable guidance and support throughout the completion of this thesis. Her expertise and insights have been instrumental in shaping my research and helping me navigate the challenges of the thesis writing process. I would also like to thank Elizabeth DaMaren for her guidance in developing my thesis as well as mentoring my interests in academia. Additionally, I extend my thanks to the Ready Lab team, and to Lucas Crupi in particular, for their support and assistance with programming. Additionally, I'd like to thank Felix Deng for his insight and support in developing interfaces with Onshape. Finally, I would like to acknowledge the participants who generously gave their time and insights to contribute to this research.

Table of Contents

List of Tables

List of Figures

1 Introduction

1.1 Computer Aided Design

Computer Aided Design (CAD) programs have had an increasing presence in the design and development of modern technologies and services. Today, it is an indispensable skill within the toolbox of modern engineers. This necessity for proficiency in CAD is evident by the increase in employers requiring CAD competency in graduating engineers [1]. Literacy in CAD is increasingly being taught within engineering programs [2], as educators strive to offer students more opportunities to gain hands on experience with modern design tools.

1.2 CAD Research

Educators of CAD continuously look to research for ways to enhance the efficacy of their teaching and ensure that students are prepared with the necessary modelling skills to excel in engineering and related industries. Despite this, an inquiry into CAD pedagogy reveals that there is no single agreed upon method on which to assess a student's understanding of CAD concepts. Various metrics and evaluation tools are available for assessing CAD skills [3], ranging from multiple-choice tests to performance-based evaluations, such as design projects or project-based learning [4]. However, assessment of CAD competency remains a challenge, as different evaluation tasks may vary in their areas of assessment, skill level, complexity, and applicability to real world scenarios. Moreover, students may respond differently to an evaluation task based on their preexisting skill levels within CAD and other technologies [5].

The absence of a universally agreed-upon standard for evaluating CAD competency further complicates the assessment process, leaving educators with the task of determining the most appropriate evaluation method for their particular context. Thus, while CAD education has become increasingly prevalent in engineering programs, it is crucial for educators to remain vigilant in their selection and application of CAD evaluation tasks to ensure that their students are being adequately prepared for the demands of the modern workplace.

1.3 Motivation and Approach

This paper attempts to develop a rigorous, efficient, and standardized protocol for evaluating a student's skill in using CAD programs. This protocol would allow educators and researchers to efficiently evaluate skill levels in users to determine the impact of a CAD activity, evaluate individual skill levels in relation to team skill levels within a group activity, and provide a quick assessment of CAD skill levels in students to adapt CAD curricula based on their needs. Furthermore, the development of an evaluation task using this framework will allow for quick and efficient assessment of CAD competency, leading to the identification of how CAD users of varying skill types respond to a given evaluation task.

The objective of establishing a framework for CAD assessment is dependent on a literature review of existing methodologies and best practices for assessing CAD skill. This review will enable the development of CAD framework that will enable users to demonstrate competency in CAD across a variety of knowledge types. The proposed framework will outline the data sources, evaluation tools and metrics used based on a user's completion of a CAD experimental task. The CAD experimental task will allow for quick and efficient assessment of CAD competency by evaluating a user's experience across a variety of skill levels.

2 Background and Literature Review

2.1 Dimensions of CAD Knowledge

Chester et al. emphasizes that CAD proficiency is spread across competency in several knowledge types, namely declarative or "command" knowledge, procedural knowledge, and strategic knowledge [6]. Declarative knowledge is knowledge about the commands existing within a CAD software. It refers to an understanding of what tools are available in the CAD tool and where they are located. Procedural knowledge is knowledge about how to construct a CAD model. It is the foresight to know what features and steps need to be taken to capture the design intent of a task. It also describes the knowledge of knowing how to execute a command in CAD software (complete an extrusion, create a patterned feature). Strategic knowledge describes the metacognitive practices that go into completing a CAD task; this involves planning, monitoring, and revising strategies that allow CAD models to be "flexible".

Ozturk et al. frames these dimensions of CAD expertise along a spectrum of "adaptive" and "routine" experts [7], where routine CAD users simply learn to perform a skill quickly and possess strong declarative knowledge, while adaptive users leverage procedural thinking and metacognition of their actions and implications on the model.

It is seen that expert CAD users are differentiated from novice users by their competency in strategic, and not declarative, CAD knowledge [6]. Because of this, procedural and strategic knowledge are identified as key skills for a high skill level in CAD [8], and are the dimensions of CAD knowledge educators are most interested in understanding. Moreover, competencies in procedural and declarative knowledge are transferable across varying CAD programs, whereas declarative knowledge is normally restricted to expertise in one particular program [9].

2.2 Existing Literature on CAD Skill Evaluation

2.2.1 Evaluation Tasks

Within the area of CAD research, evaluation of CAD skill in users is generally done through the completion of a CAD exercise or task designed to stimulate the different types of CAD knowledge one may possess. Most CAD evaluation tasks are completed using a modelling software hosted on a personal computer. The type of CAD modelling software used in evaluation tasks can vary, from uniquely custom platforms like *Energy3D* [10] to industry standards like SOLIDWORKS, CATIA, or Autodesk. A variety of these CAD evaluation tasks exist and can fall into two categories: “routine” exercises ask users to simply recreate apart from a provided reference, such as a design drawing, whereas “flexible” exercises are more abstract in nature and often do not have a direct instruction for the user to follow, but rather a set of objectives in which they may aim to meet [11]. Consequentially, routine exercises are most often deployed as surveyors of one’s declarative knowledge, and the abstractness of flexible exercises are designed to test one’s procedural and strategic knowledge [12].

Beyond the recreation of static model geometry, evaluation tasks for formula-driven modelling have also been deployed. Irwin studied the capabilities of engineering students in creating formula-driven wind turbine blades, who’s model geometry could be modified by changing model formula depending on the user’s needs [13].

2.2.2 Analytical Frameworks for CAD Environment Classification

Gopsill et al. proposed a framework to categorize the actions taken by CAD designers into six command types: creating, editing, deleting, reversing, viewing, and other [14]. This framework is useful for organizing and analyzing user design actions as well as revealing design intent.

Table 1: Action Type Classification

Action Type Name	**Creating**	**Revising**			**Viewing**	**Other**
		Editing	**Deleting**	**Reversing**		
Summary of Sample Actions	Add a sketch/ Part Studio feature/ Assembly feature Add a part from Part Studio in Assembly	Edit a sketch/ Part Studio feature/ Assembly feature	Delete a sketch/Part Studio feature/ Assembly feature Delete a part in Assembly	Redo/ undo/ cancel an operation	Open/close a tab Call animate actions	Create/delete/ rename a tab Create/merge version/branch

Table 2: Design Space Classification

Design Space	Constructive Actions				Organizing Actions	
	Part Studio		Assembly			
Action Type Name	**Sketching**	**3D Features**	**Mating**	**Visualizing**	**Browsing**	**Other Organizing**
Summary of Sample Actions	Add/ modify a sketch Copy/ paste a sketch	Add/edit a Part Studio feature * Delete a sketch/Part Studio feature	Add/delete a part from Part Studios Insert/edit/ delete an Assembly feature	Drag parts/ workspace Call animate actions	Create/delete/ rename a tab Open/close a tab	Create/merge version/branch ** Undo/redo/ cancel an operation

* Deleting a sketch is classified under 3D Features-related actions because a sketch is considered to be a part studio feature in Onshape Analytics once it is created.
** Undo/redo/cancel operations are included under Other Organizing actions because they are recorded unlinked from design spaces.

Likewise, Deng et al. developed a classification system for the design space within Onshape separating constructive actions from behavioral ones [15]. Specifically, they defined "Part Studios" as the environment where solid models are created and associated actions take place, such as sketching and extruding, while "Assemblies" refer to the space where multiple parts are connected and constrained to form systems of parts. These frameworks provide valuable tools for segmenting and categorizing the actions and design spaces used by CAD designers, which can assist in data analysis and provide insights into the design process.

2.2.3 Methods for Studying CAD Skill

An equally significant step to the evaluation task in CAD skill assessment is the method in which the completed CAD activity is scored. One of the most direct ways of identifying CAD skill is through a direct investigation of a user's model produced in CAD. Generally, this is done by the instructor of the activity, who can cross-reference a user submission with a datum model as "ground-truth". While this form of assessment can be effective, it becomes time consuming as the number of models to be checked increases. Consequentially, automated evaluation tools have been created, where a model is parsed and evaluated against set scoring criteria created by the instructor [16]–[18]. By automating the grading process, the assessment of CAD models is simplified and can be applied to models of significant size and complexity. Moreover, these CAD programs can utilize API interfaces to directly interact with the CAD model and create changes, validating the ability for CAD models to be adaptable [19]. To some extent, entire CAD courses have been automated to provide objective and repeatable feedback on student models quickly [20].

There also exists a tangible form of CAD skill assessment in the use of self-graded assessment rubrics. These rubrics can be distributed as surveys before and after a CAD activity to judge self-perceptions of CAD skill before and after completing a set of tasks. A common issue seen with these surveys is that these rubrics do not necessarily yield a valid judgement of one's performance [21], and there is a discrepancy in perceived-skill and instructor assessment score. Company et al. attempts to bridge the gap between these two and improve the accuracy of self-assessment through adaptable rubrics enriched with dynamic resources that improve understanding [22], [23].

Finally, there have been attempts at capturing CAD skill not seen through explicit interaction with a CAD program. Researchers have experimented with the use of think-aloud exercises that have students verbalize their though process to describe the modelling strategies they would use to create a model of a part [24] [25]. One problem with this form of assessment is the time-consumption required to verbally describe the modelling process, which can increase as model complexity increases. Tangibly, Rynne et al. studied the design intent of CAD modelers through their cognitive actions by developing a framework to define cognitive part modelling tasks and intervention methods to support efficient uses of parametric-modelling [26]. Daud et al. evaluated the conceptual knowledge of mechanical engineering students in CAD using concept maps that visualized how the students represented their knowledge structure in CAD depending on the domain or modelling task presented [27].

2.2.4 CAD User Traits and Model Attributes

Within the context of a CAD evaluation task, several metrics have been studied. Each metric can reveal a user's grasp of a specific dimension of CAD knowledge. Within the dimension of declarative knowledge, Hamade uses the "Time-to-Completion" (TTC) as an indicator of how well a CAD user can interface with the CAD software and perform modelling operations [8]. Those with high command knowledge would be able to complete CAD actions quicker and subsequently lower their TTC.

Similarly, the transition rate between design actions can be an important factor in understanding the skill level of students in CAD. By analyzing the sequence and frequency of actions taken by CAD users, it is possible to gain insight into their proficiency with the software and competency in modelling knowledge [28].

Moreover, correctness of a CAD model is also an indicator of declarative knowledge, as it reveals how well the user can carry out a design task independent of the process used. Chen performed a CAD experiment on novel and expert engineers by recording how accurate their actions were, that is, how well they aligned with the correct answer, as well as how complete their models were when compared to the task [29].

Recent studies have looked at the relationship between procedural knowledge and model complexity. Hamade, for example, analyzed the feature complexity of each model and the number of steps taken to create a particular geometry, suggesting that those with greater procedural knowledge can use less extensive, but more complex modeling tools to create the geometry in fewer steps [8]. Ault et al. developed an analytical equation to define the complexity index (CI) of a part as a numerical value, which is the sum of features and sketch entities used to complete a given CAD task [30]. The CI is often used as an indicator of model complexity independent of the design process. In a recent study, Xie proposed the creation/revision ratio as another indicator of procedural knowledge, which measures how iterative a user may be through the ratio of creating CAD actions to revising ones [31].

$$CI = \sum_{i=0}^{F_S} E_i + \sum_{i=0}^{F_E} \sum_{j=1}^{S} N_{ij} + \sum_{i=0}^{F_H} C_i + \sum_{i=0}^{F_M} CI_i + \sum_{i=0}^{F_P} CI_i$$

(Eq. 1)

Figure 1: Complexity Index Formula

3 Framework

3.1 Overview

Understanding that the evaluation of a user's CAD skill is dependent on the knowledge dimension, evaluation method, data collection, and attributes, the following framework is presented as a protocol to evaluating CAD skill. The framework utilizes data from both the CAD software itself and the presented CAD model. From here, the data is parsed into a collection of datasets classifying the data based on the action type and design space environment. This refined description and filtering allows for metrics to be calculated that describe the level of understanding a user has with the CAD knowledge dimensions.

Overall, the framework provides a structured and objective approach to evaluating CAD skill that is based on a thorough analysis of relevant data that reveals a user's competency in CAD through several metrics and analysis methods developed through the literature. An overview of the framework is presented in Figure 2.

3.2 Data Inputs

Backend analytical data is directly exported from the CAD model. This data describes the type of design actions the user is taking, as well as what design space they are working in. This is complimented by the CAD model provided by the user, which provides data on the model itself, including the types of features and sketch entities that exist within it. Moreover, data input from the user in the form of auditory and video recording of interfacing the CAD software in think-aloud exercises is also captured.

3.3 Evaluation Methods

Three tools are proposed to parse through the data. Backend analytical data is taken, and an analytical framework is used to classify actions based on their action and design space. Action type describes the type of action the user has taken, such as creating, deleting, viewing, and editing, while the design space describes what environment the user is modelling in, such as an assembly or part environment. The CAD model itself is evaluated using an automated tool that captures the types of features used like extrusions, revolves, fillets, as well as the total number of features a model has.

3.4 Metrics of Analysis

The following metrics are derived from the data filtered by the evaluation tools. Time-to-completion is derived from the backend analytical data. The creation/revision ratio and transition rate are derived from the action type and design space classifications respectively. The CAD model reveals the feature count, completeness of the model and accuracy, as well as the complexity index describing the efficiency of the model to use features effectively.

3.5 Results

The results of the framework can provide insight into a user's CAD skills and attributes. The creation/revision ratio can reveal how iterative a user is in their design process, based on how many editing or creating actions exist in their modelling process. The transition rate can reveal how reflexive the user is in their modelling practice based on how often they utilize a variety of CAD actions and access design spaces. The time-to-completion can assess how efficient a user is at completing a CAD task in a timely fashion. The feature count and complexity index can assess how efficient is a user in the way they recreate model geometry through their use of features. Moreover, their ability to progress through CAD tasks and their accuracy can describe how accurate they are in following a CAD task.

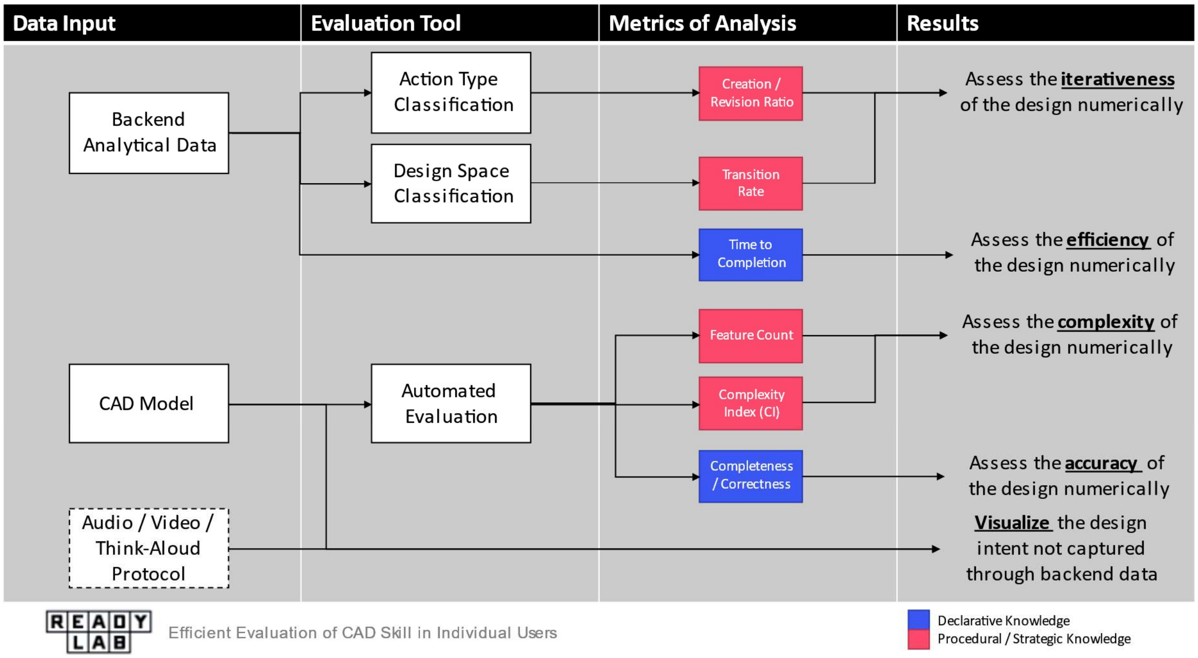

Figure 2: CAD Skill Evaluation Framework

4 Case Study of an Individual CAD Exercise

4.1 Overview

To evaluate the feasibility of the proposed framework, a case-study was developed that tasks users to perform several CAD modelling tasks. The case-study was deployed to a first-year engineering class at the University of Toronto, and the participants completed the activity on their own time. The participants also filled in a survey that attempted to capture their self-perceived CAD skill prior to completing the case-study. The experimental tasks involved creating various CAD geometry of varying difficulty within a specified time limit, as well as making changes to the models created in the first task.

4.2 Case Study Goals

The design of the case-study should support the evaluation of the proposed framework by providing an accessible method to studying CAD competency in students. The following list describes the goals set for the design of the experimental task:

- The modelling task difficulty should be appropriate for a variety of skill levels.

- The modelling task difficulty should increase as the user progresses through the experiment.
- The CAD geometry used in the modelling task should allow for a variety of modelling strategies.
- The time allotted for each CAD task should be 20 minutes.

4.3 Case Study Design Tasks

The CAD task involves four phases. Phase 1 requires the user to recreate a simple CAD geometry. Phases 2 and 3 requires the user to build off their model in Phase 1 through the inclusion of additional features. Phase 4 requires the user to take their developed model and make changes to it. Phases 1 to 3 are designed to evoke the routine expertise of users through a "muscle-memory" CAD activity while Phase 4 pulls on users to demonstrate their procedural and strategic knowledge through an adaptive exercise. The increasing complexity of each design task can reveal the differences in design intent between users of varying skill levels. The CAD geometry used provides the opportunity users with a variety of potential modelling paths to follow and the ability to leverage feature operations like patterns and symmetry. Figure 14 to Figure 17 in Appendix A illustrate the CAD geometry provided to the participants in each phase of the case-study. Each user had an allotment of 20 minutes to complete each phase, for a total of 80 minutes to finish the entire exercise.

4.4 Onshape CAD Software

The case-study was deployed using Onshape's CAD software. One of the key benefits of using Onshape for a CAD case-study is its ability to provide backend data for user analysis. This data includes information on a user's modeling history, design iterations, and collaboration activity, which can be used to evaluate their CAD skills. Additionally, Onshape's cloud-based architecture allows for easy collaboration and version control. Each of the participants received a unique workspace on the Ready Lab enterprise account to participate in the study and complete their modelling tasks. The workspace included the instructions and drawings to complete the modelling exercises.

4.5 Classroom Study

The case study was shared with a first-year engineering class enrolled in the Engineering Science program at the University of Toronto. The focus group consisted of a group of students enrolled in Praxis II: a foundational design course introducing students to engineering design models and tools as well as teamwork and professionalism.

5 Discussion

5.1 Post-Processing of Case Study Data

With approval from the REB and consent from participants, the CAD designs created in Onshape and backend data were used to evaluate the feasibility of the evaluation framework. In total, 117 students consented to the study and 11 attempted the exercise. Five exercises were omitted due to a lack of any significant progress in the model made beyond opening the activity instructions, for a total of six models available for review. In this case-study, 2329 design actions were recorded by the participants.

5.2 Audit Trail Extraction from Onshape

Detailed data of user actions are exported through Onshape in the form of audit trails. The audit trails are chronological sequences that list all analytical actions performed by the user within their workspace. Each audit trail entry includes the timestamp, document name, element tab, user, and description of the action taken. The audit trails are exported through OnShape as a CSV file. The data presented by the audit trails requires some cleaning, as the data set includes some redundant or duplicate entries depending on the actions taken by the user. Additionally, some “translating” is required to convert the descriptions produced by the audit trail entries for data analysis.

	Time	Document	Tab	User	Description
1	3/20/2023 3:26	CAD Research Task Version B (UofTW23) - 128	N/A	[redacted]	Close document
2	3/20/2023 3:26	CAD Research Task Version B (UofTW23) - 128	Task Part	[redacted]	Tab Task Part of type PARTSTUDIO closed by [redacted]
3	3/20/2023 3:26	CAD Research Task Version B (UofTW23) - 128	Task Part	[redacted]	Changes saved when Onshape closed.
4	3/20/2023 3:26	CAD Research Task Version B (UofTW23) - 128	Task Part	[redacted]	Add or modify a sketch
5	3/20/2023 3:26	CAD Research Task Version B (UofTW23) - 128	CAD Research Task Instructions - UofT.pdf	[redacted]	Tab CAD Research Task Instructions - UofT.pdf of type BLOB closed by [redacted]
6	3/20/2023 3:26	CAD Research Task Version B (UofTW23) - 128	Task Part	[redacted]	Change Element Timings Event
7	3/20/2023 3:26	CAD Research Task Version B (UofTW23) - 128	Task Part	[redacted]	Tab Task Part of type PARTSTUDIO opened by [redacted]
8	3/20/2023 3:25	CAD Research Task Version B (UofTW23) - 128	Design Activity 1.pdf	[redacted]	Tab Design Activity 1.pdf of type BLOB closed by [redacted]
9	3/20/2023 3:25	CAD Research Task Version B (UofTW23) - 128	CAD Research Task Instructions - UofT.pdf	[redacted]	Tab CAD Research Task Instructions - UofT.pdf of type BLOB opened by [redacted]
10	3/20/2023 3:25	CAD Research Task Version B (UofTW23) - 128	CAD Research Task Instructions - UofT.pdf	[redacted]	Change Element Timings Event

Figure 3: Audit Trail Entries (usernames redacted)

5.3 Automated Evaluation of CAD Models

Evaluation of the CAD models was done by creating a custom program to evaluate the evaluation tools and metrics presented in the framework. The program is written in python and leverages the audit trails and API interface within Onshape. The API interface in Onshape is a programming interface that allows developers to access and manipulate CAD data, including metadata about designer actions. The API is used in the program to extract feature information of the CAD models made by the students to compute the metrics proposed in the framework, as well as confirm if the model is constructed correctly according to the drawings provided.

Metrics, such as the time to completion and transition rate, were calculated with a python program that parsed the audit trail entries for the first and last actions for each user at each phase of the case-study. The rest of the metrics in the framework were calculated using feature data returned from the Onshape API interface.

```python
def featureCheck(self):

    for key in self.URL:
        self.features[key] = self.api_request(self.my_keys, self.url_prefix,
                                              '/api/partstudios/d/' + self.identifiers[key][0] + '/w/' +
                                              self.identifiers[key][
                                                  1] + '/e/' + self.identifiers[key][2] + '/features', {}, 'get')

        # Count the number of different feature types
        sketch_entities = {}
        extrusions = {}
        fillets = {}
        chamfers = {}
        cuts = {}
        revolves = {}
        sweeps = {}
        lofts = {}
        linear_patterns = {}
        circular_patterns = {}
        feature_data = self.features[key]
        feat = {}

        for feature in feature_data['features']:

            if key not in feat:
                feat[key] = {"sketch_entities": 0, "extrusions": 0, "edge_features": {"fillets": 0, "chamfers": 0},
                             "cuts": 0, "revolves": 0, "sweeps": 0, "lofts": 0, "linear_patterns": 0,
                             "circular_patterns": 0}
            feature_type = feature['message']['featureType']

            if feature_type == 'newSketch':
                feat[key]['sketch_entities'] += len(feature['parameters'])
            elif feature_type == 'extrude':
                feat[key]['extrusions'] += 1
            elif feature_type == 'fillet':
                feat[key]['edge_features']['fillets'] += 1
            elif feature_type == 'chamfer':
                feat[key]['edge_features']['chamfers'] += 1
            elif feature_type == 'CUT':
                feat[key]['cuts'] += 1
            elif feature_type == 'revolve':
                feat[key]['revolves'] += 1
            elif feature_type == 'sweep':
                feat[key]['sweeps'] += 1
            elif feature_type == 'loft':
                feat[key]['lofts'] += 1
            elif feature_type == 'linear_pattern':
                feat[key]['linear_patterns'] += 1
            elif feature_type == 'circular_pattern':
                feat[key]['circular_patterns'] += 1
```

Figure 4: Code Snippet of Python Program

5.4 Results of Case-Study

5.4.1 Design Space and Action Type Classification

Upon examining Figure 6, it is apparent that most participants' action types are centered around creating and viewing models, while fewer users perform editing or deleting actions. This is seen in Users 831, 664, and 627, who completed almost no edits. Taking a closer look at the audit trails, a behavior pattern of creating-reversing or creating-deleting was observed. This tendency to delete or reverse rather than revise unsatisfactory elements may be an indicator of a lack of awareness regarding the command interface of the CAD software, and subsequently a low level of declarative knowledge [32]. Figure 5 illustrates how User 128 created a sketch and attempted to add a part-studio feature, only to undo all their work and start over possibly because they recognized an error in their process.

3/20/2023 3:23	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Commit add or edit of part studio feature
3/20/2023 3:23	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Edit : Sketch 5
3/20/2023 3:23	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Add or modify a sketch
3/20/2023 3:22	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Start edit of part studio feature
3/20/2023 3:22	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Commit add or edit of part studio feature
3/20/2023 3:22	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Add or modify a sketch
3/20/2023 3:22	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Edit : Sketch 5
3/20/2023 3:21	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Undo Redo Operation
3/20/2023 3:21	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Undo Redo Operation
3/20/2023 3:20	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Undo Redo Operation
3/20/2023 3:20	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Undo Redo Operation
3/20/2023 3:20	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Undo Redo Operation
3/20/2023 3:20	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Undo Redo Operation
3/20/2023 3:19	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Undo Redo Operation
3/20/2023 3:19	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Undo Redo Operation
3/20/2023 3:18	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Start edit of part studio feature
3/20/2023 3:18	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Commit add or edit of part studio feature
3/20/2023 3:18	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Add or modify a sketch
3/20/2023 3:18	CAD Research Task Version B (UofTW23) - 128	Task Part	ID 128	Insert feature : Sketch 5

Figure 5: Audit Trail Snippet of User 128 Modelling Actions

User 851 and 627's large number of creation actions may suggest that they were attempting to learn the command interface of Onshape by trying out different features. Another trend, as seen in Figure 5 is that users prefer to "undo" any errors in their work rather than making revisions, as evidenced by the high number of reversing actions in users 128, 851, and 627. However, this approach can be risky if mistakes are discovered later in the modelling process, as significant progress may need to be undone to address the issue. Notably, user 851 had a substantial number of reversing actions, indicating that they attempted to use certain features in their CAD model several times before ultimately undoing them due to compatibility issues with future features. The prevalence of reversing actions may also suggest that some users have lower levels of procedural knowledge, as they may not fully understand how certain features will impact the overall modelling process. In the case of user 851, their repeated attempts to use a specific feature before ultimately undoing it further support this hypothesis. Users are also taking a large number of viewing actions, possibly to gain a further understanding of the task and the drawings. Few users had editing actions except for user 128, who had almost as many edits as creation actions. Many edits may suggest greater procedural knowledge, and possibly a more flexible model, if only edits are needed to respond to design changes rather than creating new entities and features.

In Figure 7, it can be inferred that most users spent their time browsing tasks and reviewing design drawings in the activity instructions. It is also observed that users frequently switch between the activity instructions and the modeling environment, which may suggest that the reference models were too difficult. Moreover, it was found that there were more 3D modeling actions taken than sketching, which could be due to the reference model requiring many 3D features like holes, chamfers, and extrusions compared to its sketching being largely prismatic and constructed with simple elements like rectangles and squares.

Additionally, it was observed that users referred to the model drawings for later phases of the exercise, despite this being discouraged in the activity instructions. One possible explanation for this behavior is that users anticipated that each successive phase would build on the existing geometry, and therefore, they were curious to see how future models could look and utilized this foresight in finishing the existing task.

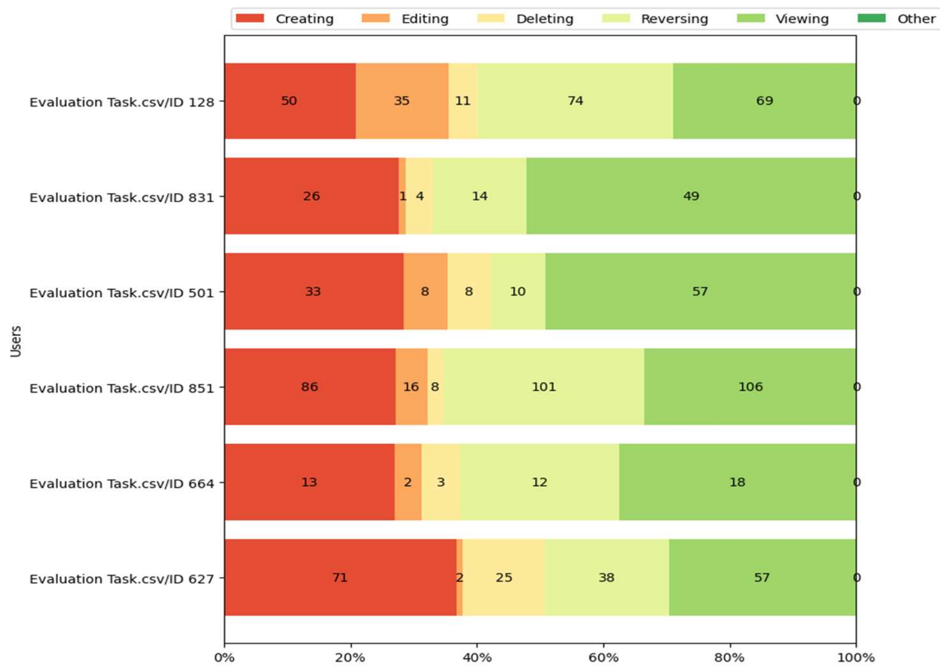

Figure 6: Action Type Classification

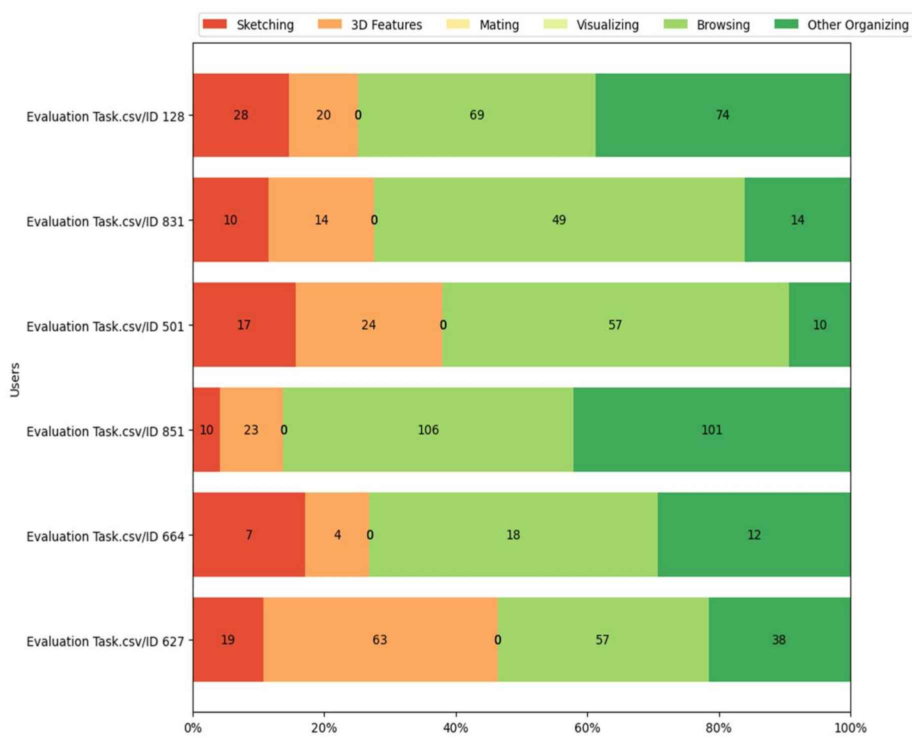

Figure 7: Design Space Classification

5.4.2 Time to Completion

Of the six participants took part in the study, all of them completed Phase 1. However, only 4 out of 6 participants completed Phase 2, while one user managed to complete Phases 3 and 4. For participants who attempted but could not finish an activity, their completion time was omitted from the analysis.

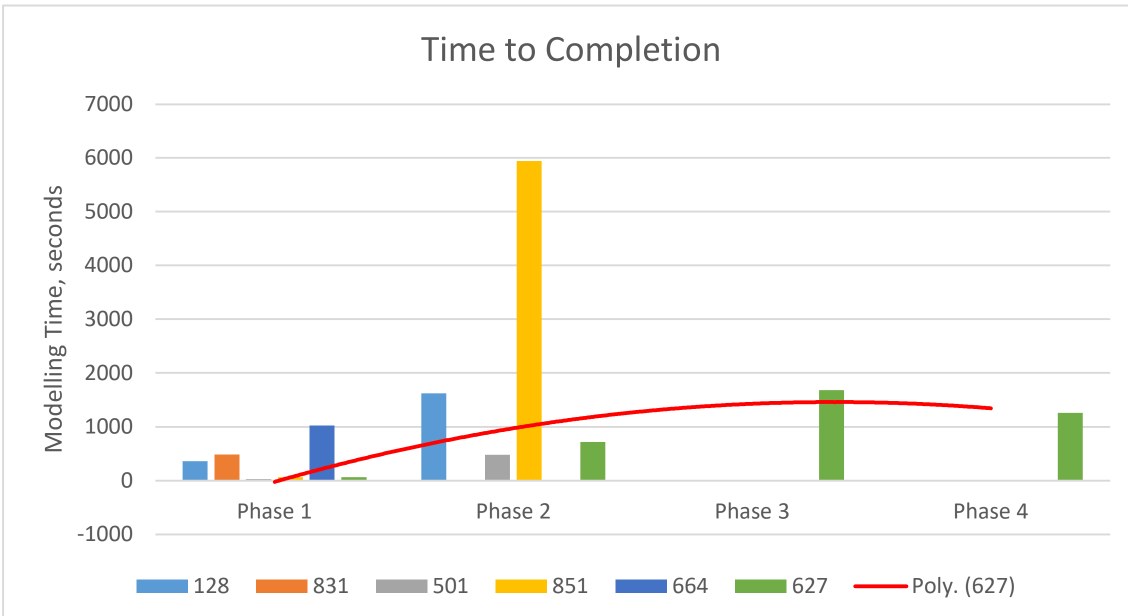

Figure 8: Completion Time for each Task

Looking at Figure 8, it can be observed that most users completed Phase 1 quickly and within the time limit, except for user 664 who spent a significant amount of time viewing activity drawings of subsequent tasks before attempting to model the geometry in Phase 1. Phases 2 through 4 show an increase in modeling difficulty, as indicated by the longer time to completion and lower completion rates. In Phase 3, only one user was able to complete the task, albeit exceeding the allotted time of 20 minutes. However, this user was able to make the necessary changes within the allocated time in Phase 4, indicating that the flexibility of their model made it easier for them to make changes downstream. It is important to note that many users went beyond the 20-minute time limit in each of the phases of the activity, possibly suggesting that the model difficulty was too much for the participants, or that there could have been a lack of declarative knowledge necessary to navigate the software in a timely fashion.

5.4.3 Transition Rate

Based on the analysis of Figure 9, it can be observed that the number of transition actions varies across different users as they switch between design spaces and action types. User 644 had the lowest number of transition actions, which can be attributed to their higher number of browsing and viewing actions. On the other hand, users 128, 851, and 627 had a larger number of transitions, revealing a greater variety of actions in their modelling strategies. Furthermore, the users with a higher number of transitions also had a greater spread in the composition of their design space and action type classifications. This suggests that they used a wider range of CAD actions in different modelling environments.

The transition ratio in design spaces and action types appears to be nearly equal for all participants, indicating that all users switched between design space and action type classifications equally. An exception to this is user 851, who's transition ratios were identical for design space and action type classifications.

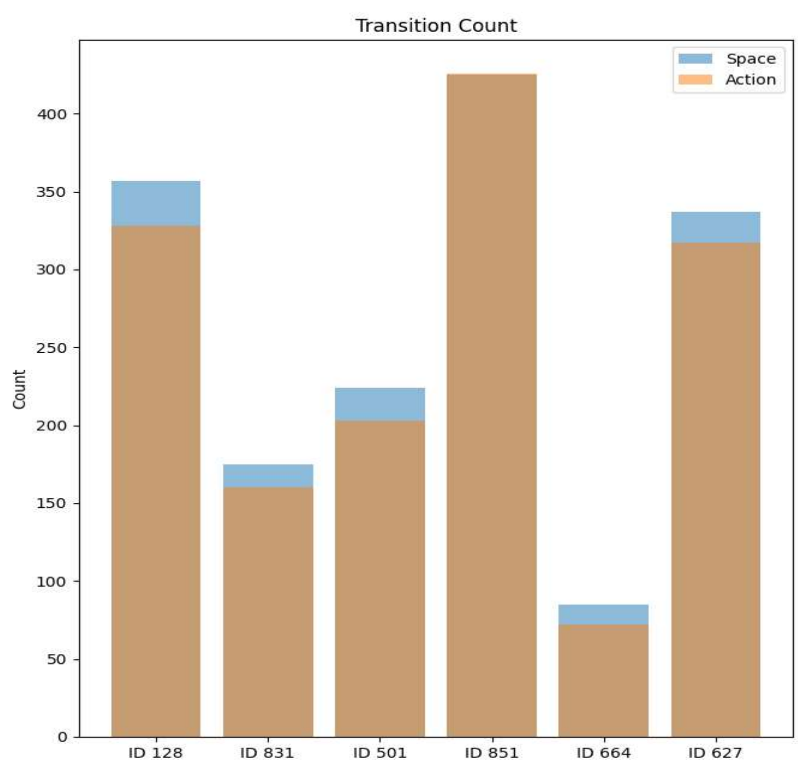

Figure 9: Number of transitions between design spaces and actions

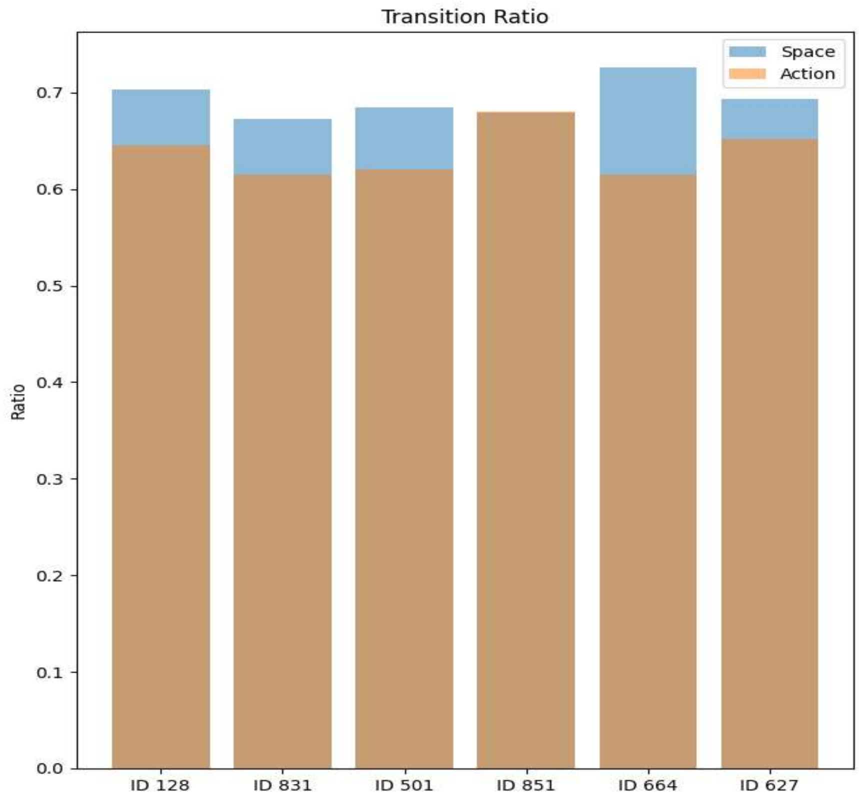

Figure 10: Ratio of Transitioning Actions to Total # of Actions

5.4.4 Creation/Revision

The creation/revision ratio can reveal a better understanding of the design behaviors and skill levels of the participants. Figure 11 shows the creation/revision ratio for each participant across the entire activity. The results indicate that the participants each possess a different modelling strategy: user 128 had the lowest C/R ratio of 0.28, which is evident in the large number of reversing actions they made. A lower C/R ratio could also be an indicator of how well a user was at modelling the correct geometry on the first pass, with more and more refinements suggesting a less-confidence approach [29]. Users 831, 501, and 627 each had a value greater than one, indicating that they spent more time creating new features rather than revising them.

It is important to note that each user's progression in the case-study would have an impact on their C/R ratio. This is because the first three phases of the case-study are primarily focused on creation, while the last phase is exclusively an iterative exercise.

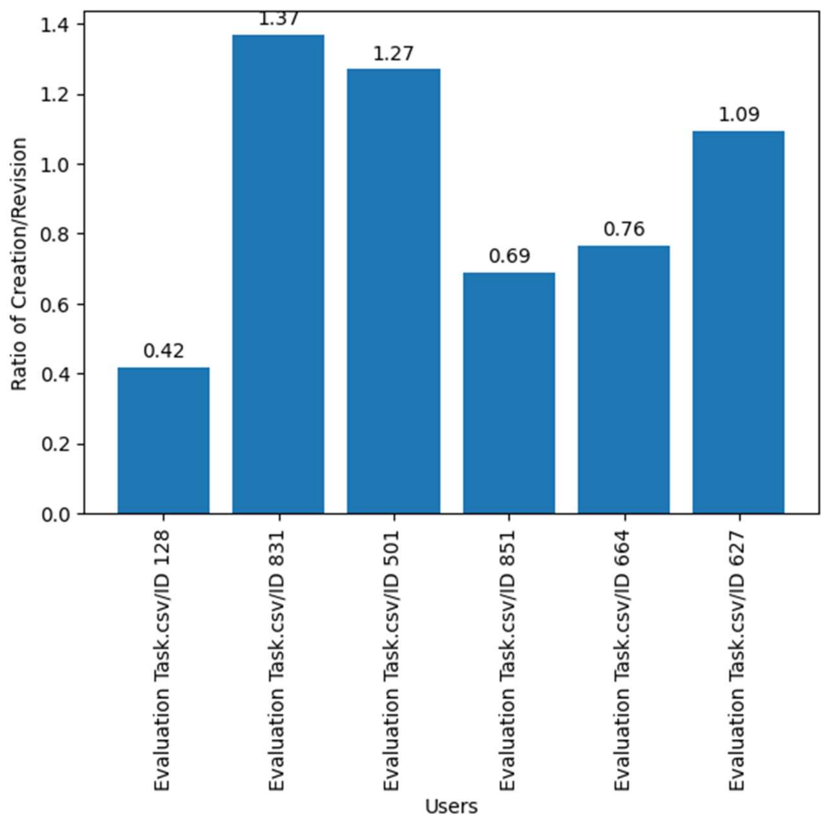

Figure 11: Creation / Revision Ratio of Users

5.4.5 Feature Count & Complexity Index

The complexity index (CI) was calculated for each participant at every phase of the CAD modelling task to assess their strategic planning and problem-solving skills. As expected, all participants had the same CI score for the first phase, as the geometry was straightforward enough for each user to quickly visualize a modelling plan and execute the same commands. However, in the second phase, there was a distinct difference in CI scores. Users 128, 501, and 627 had CI scores of 13, 14, and 15, respectively, whereas user 851 had a significantly higher score of 25, indicating a difference in the user's strategic planning during modelling. Examining the models, it was found that User 851 had left several unused sketches in their model, which had a negative impact on their CI score. This could support the idea that users with greater CAD expertise will require less sketching to complete a given task [32]. In Phase 2, User 627 had created a new edge feature for each edge of their model rather than selecting multiple faces in one feature, resulting in twice as many edge features than the rest of the group. This design choice would have had consequences later during the modifying phase, as they would have had to change each fillet feature individually rather than just one, subsequently increasing their CI score. Few users utilized the hole feature, instead opting to deselect the circular geometry when extruding the base of the part, saving the need for an extra hole feature.

	Phase 1											Phase 2											Phase 3											Phase 4										
User ID	FS	EI	FE	S	NIJ	FH	CI	FM	FP	CI	**SCORE**	FS	EI	FE	S	NIJ	FH	CI	FM	FP	CI	**SCORE**	FS	EI	FE	S	NIJ	FH	CI	FM	FP	CI	**SCORE**	FS	EI	FE	S	NIJ	FH	CI	FM	FP	CI	**SCORE**
128	1	4	0	0	0	0	0	0	0	0	**4**	2	10	2	0	0	1	1	0	0	0	**13**											**0**											**0**
831	1	4	0	0	0	0	0	0	0	0	**4**											**0**											**0**											**0**
501	1	4	0	0	0	0	0	0	0	0	**4**	3	12	2	0	0	0	0	0	0	0	**14**											**0**											**0**
851	1	4	0	0	0	0	0	0	0	0	**4**	3	23	2	0	0	0	0	0	0	0	**25**											**0**											**0**
664	2	4	0	0	0	0	0	0	0	0	**4**											**0**											**0**											**0**
627	1	4	0	0	0	0	0	0	0	0	**4**	3	11	4	0	0	0	0	0	0	0	**15**	8	30	4	0	0	0	0	0	0	0	**34**	8	28	4	0	0	0	0	0	0	0	**32**

Figure 12: CI Calculation

The CI score for each participant increased as the model became more complex. User 627, who completed phases 3 and 4, experienced a plateau in their CI score at the last routine modelling task of the study, before dropping slightly during the adaptive phase that required them to make changes. This drop was likely due to the participant recognizing unutilized sketches in their model and subsequently cleaning up their process by reducing the number of sketch entities used during phase 4. This observation could suggest that a user's strategic thinking is affected when prompted to iterate on a CAD model, as the challenge of revising and editing the model prompts them to check the entire model for errors. The relationship between the complexity index and model complexity shares a similar trend to the Time to Completion observed earlier, with an increasing in both as model complexity increased before plateauing at the revising phase of the case-study.

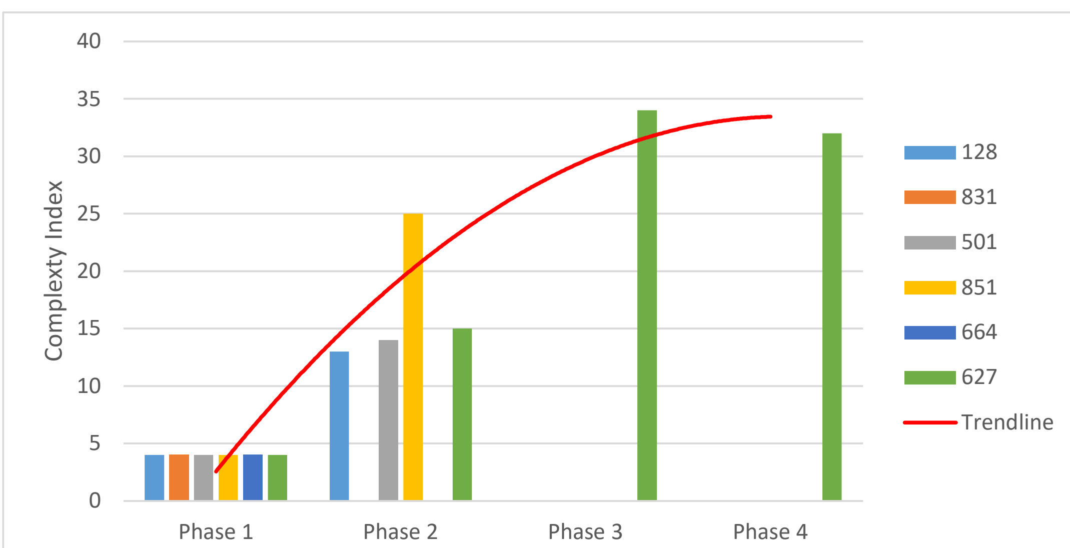

Figure 13: Complexity Index of Case-Study Participants

6 Conclusions and Future Work

6.1 Conclusions

In conclusion, this paper presented a comprehensive framework for evaluating CAD skill levels in students. Through a thorough literature review, various evaluation tools, data classifications, knowledge dimensions, and model attributes were analyzed and synthesized into a protocol that leverages both backend analytical data and tangible verbal/auditory data to classify a user's modeling behavior based on design space and action type. This classification then allows for the calculation of metrics that provide insight into a user's grasp of declarative, procedural, and strategic knowledge types.

The effectiveness of this protocol was demonstrated through a case study involving six participants enrolled in a first-year undergraduate engineering program. The study revealed that many users struggled with the presented tasks, spending a great deal of time performing viewing and visualization actions, as well as going over the allotted time for each task. This suggests that the tasks were beyond their skill levels. Furthermore, the participants spent a significant amount of time attempting to learn the Onshape software interface through a large number of creating and revising actions, which impacted their design behavior and design intent.

The results of the framework showed that many users opted to undo and delete their mistakes in modelling rather than revise them. The C/R scores of the participants were subsequently affected by the large number of editing actions made. Participants also displayed many transition actions as they switched between modelling and viewing the activity instructions. Additionally, the feature count for each participant increased as model complexity increased. Design tasks that prompted the user to modify their designs resulted in a lower feature count as they made changes to earlier inefficient modelling practices. The Time to Completion of the participants also increased with a more difficult modelling challenge, before decreasing at the modifying phase.

Overall, this study highlights the importance of evaluating CAD skill levels in students to ensure they are adequately prepared for real-world applications. This protocol provides a valuable tool for educators to assess and develop students' CAD competencies. However, future studies could benefit from exploring the effectiveness of this protocol in other educational settings and with larger sample sizes. Ultimately, the insights gained from this research can inform the development of more effective CAD training programs and improve the quality of CAD designs in engineering and related fields.

6.2 Limitations

Several limitations exist in the presented work. One of which is a lack of information regarding the participants pre-experiment survey responses. Self-reported CAD expertise levels could have enriched the analyses made in the case-study by comparing their perceived CAD skill with their actual performance. It was not known if the participants had previous experience with Onshape or any CAD program at all. The presented activities in the experiment did not allow for much flexibility, and future experiments could provide a more open-ended project with design objectives for the participants to demonstrate procedural knowledge.

While parsing the audit trails, it was seen that they lacked the resolution to describe certain design actions, such as the processes occurring during sketches, which may have revealed greater detail into what types of creating and editing actions participants would use in the sketching environment.

Based on the data collected from the case-study, it appears that participants may be struggling to complete the design phase within the allotted time of 20 minutes. This can lead to a negative experience for participants and may impact the quality of the designs produced. To address this issue, it may be beneficial to consider extending the allotted time for the design phase in future iterations of the project. This would give participants more time to work on their designs and reduce the likelihood of them feeling rushed or unable to complete the task.

The API program did not cover certain design features, such as sweeps, that some of the participants ended up using in their CAD models. This meant that manual analysis of some models had to be made. A resolution would be to enrich the number of features the program would parse for when reading feature data of a CAD model.

While the presented case-study results do reveal some efficiency of the framework, it has not been evaluated against users with a known experience level in CAD, or by industry-experts. It may be useful to have the protocol deployed to a panel of expert CAD modelers, similar to a Delphi process [33] to vet the ability for the protocol to accurately display CAD skill.

6.3 Future Work

To further enhance the capabilities of the presented framework, future work could focus on several areas. One potential avenue for exploration is to develop the API interface to include more modelling features, thus allowing for a more comprehensive capture of the modelling environment. Additionally, the effects of the think-aloud exercise on revealing CAD skill could be investigated, potentially providing further insight into the relationship between verbalization and skill acquisition. Pre-study surveys may also be useful in identifying user-perceived skill levels, allowing for comparison with the results generated by the framework. More work should go into developing the tasks so that there is ample time and resources available for the participants to complete the model. These tasks could also include collaborative modelling scenarios or make use of assembly modelling, allowing for a more diverse range of evaluation criteria. Additionally, the framework should be deployed to expert CAD modelers to vet its ability to reveal the right evaluation for users of varying skill levels. Finally, future research could focus on the applicability of the framework over the duration of an entire course, potentially revealing improvements in CAD skill acquisition and learning in students over a longer period. The evaluation of CAD skill produced by this framework could be paired with a Wright's Learning Curve (WLC) to investigate the learning paths for acquiring CAD skill in different user profiles [34].

References

[1] X. Ye, W. Peng, Z. Chen, and Y.-Y. Cai, “Today’s students, tomorrow’s engineers: an industrial perspective on CAD education,” *Computer-Aided Design*, vol. 36, no. 14, pp. 1451–1460, Dec. 2004, doi: 10.1016/j.cad.2003.11.006.

[2] A. Carberry and A. McKenna, “Exploring Student Conceptions of Modeling and Modeling Uses in Engineering Design,” *Journal of Engineering Education*, vol. 103, Jan. 2014, doi: 10.1002/jee.20033.

[3] T. Delahunty, N. Seery, R. Dunbar, and M. Ryan, “An exploration of the variables contributing to graphical education students’ CAD modelling capability,” *International Journal of Technology and Design Education*, vol. 30, Apr. 2020, doi: 10.1007/s10798-019-09503-x.

[4] F. Fernandes, N. Júnior, A. Daleffe, D. Fritzen, and R. Alves de Sousa, “Integrating CAD/CAE/CAM in Engineering Curricula: A Project-Based Learning Approach,” *Education Sciences*, vol. 10, p. 125, Apr. 2020, doi: 10.3390/educsci10050125.

[5] A. Dagman and K. Wärmefjord, “An Evidence-Based Study on Teaching Computer Aided Design in Higher Education during the COVID-19 Pandemic,” *Education Sciences*, vol. 12, no. 1, p. 29, Jan. 2022, doi: 10.3390/educsci12010029.

[6] I. Chester, “Teaching for CAD expertise,” *Int J Technol Des Educ*, vol. 17, no. 1, pp. 23–35, Mar. 2007, doi: 10.1007/s10798-006-9015-z.

[7] E. Ozturk, B. Yalvac, M. Johnson, X. Peng, and K. Liu, “Adaptive Expertise and its Manifestation in CAD Modeling: A Comparison of Practitioners and Students,” in *2015 ASEE Annual Conference and Exposition Proceedings*, Seattle, Washington: ASEE Conferences, Jun. 2015, p. 26.155.1-26.155.14. doi: 10.18260/p.23494.

[8] R. F. Hamade, “Profiling the Desirable CAD Trainee: Technical Background, Personality Attributes, and Learning Preferences,” *Journal of Mechanical Design*, vol. 131, no. 12, p. 121009, Dec. 2009, doi: 10.1115/1.4000455.

[9] M. D. Johnson, E. Ozturk, B. Yalvac, L. Valverde, X. Peng, and K. Liu, “Examining Adaptive Expertise: A Novel Comparison of Student and Practicing Engineer CAD Modeling Performance,” presented at the ASME 2015 International Mechanical Engineering Congress and Exposition, American Society of Mechanical Engineers Digital Collection, Mar. 2016. doi: 10.1115/IMECE2015-50296.

[10] “A Computer-Aided Design Based Research Platform for Design Thinking Studies | J. Mech. Des. | ASME Digital Collection.” https://asmedigitalcollection.asme.org/mechanicaldesign/article-abstract/141/12/121102/956260/A-Computer-Aided-Design-Based-Research-Platform?redirectedFrom=fulltext (accessed Apr. 04, 2023).

[11] R. P. Diwakaran and M. D. Johnson, “Analyzing the effect of alternative goals and model attributes on CAD model creation and alteration,” *Computer-Aided Design*, vol. 44, no. 4, pp. 343–353, 2012, doi: 10.1016/j.cad.2011.11.003.

[12] M. Johnson, E. Ozturk, J. Johnson, B. Yalvac, and X. Peng, "Assessing an Adaptive Expertise Instrument in Computer-aided Design (CAD) Courses at Two Campuses," presented at the 2012 ASEE Annual Conference & Exposition, Jun. 2012, p. 25.212.1-25.212.18. Accessed: Nov. 12, 2022. [Online]. Available: https://peer.asee.org/assessing-an-adaptive-expertise-instrument-in-computer-aided-design-cad-courses-at-two-campuses

[13] J. Irwin, "Wind Turbine Blade CAD Models used as Scaffolding Technique to Teach Design Engineers," *Engineering Design Graphics Journal*, vol. 77, no. 2, 2013.

[14] J. Gopsill, C. Snider, L. Shi, and B. Hicks, "Computer aided design user interaction as a sensor for monitoring engineers and the engineering design process," presented at the Proceedings of International Design Conference, DESIGN, 2016, pp. 1707–1718.

[15] Y. Deng, M. Mueller, C. Rogers, and A. Olechowski, "The multi-user computer-aided design collaborative learning framework," *Advanced Engineering Informatics*, vol. 51, p. 101446, Jan. 2022, doi: 10.1016/j.aei.2021.101446.

[16] N. Bojcetic, F. Valjak, D. Zezelj, and T. Martinec, "Automatized Evaluation of Students' CAD Models," *Education Sciences*, vol. 11, p. 145, Mar. 2021, doi: 10.3390/educsci11040145.

[17] S.-H. Joo, "Assessment of Three Dimensional CAD Models Using CAD Application Programming Interface," presented at the ASME 2018 International Mechanical Engineering Congress and Exposition, American Society of Mechanical Engineers Digital Collection, Jan. 2019. doi: 10.1115/IMECE2018-87776.

[18] S. Kirstukas, "Development and Evaluation of a Computer Program to Assess Student CAD Models," in *2016 ASEE Annual Conference & Exposition Proceedings*, New Orleans, Louisiana: ASEE Conferences, Jun. 2016, p. 26781. doi: 10.18260/p.26781.

[19] K. Jaakma and P. Kiviluoma, "Auto-assessment tools for mechanical computer aided design education," *Heliyon*, vol. 5, p. e02622, Oct. 2019, doi: 10.1016/j.heliyon.2019.e02622.

[20] M. Guerci and D. Baxter, "Automating An Introductory Computer Aided Design Course To Improve Student Evaluation," in *2003 Annual Conference Proceedings*, Nashville, Tennessee: ASEE Conferences, Jun. 2003, p. 8.258.1-8.258.9. doi: 10.18260/1-2--11479.

[21] A. Jonsson and G. Svingby, "The use of scoring rubrics: Reliability, validity and educational consequences," *Educational Research Review*, vol. 2, no. 2, pp. 130–144, Jan. 2007, doi: 10.1016/j.edurev.2007.05.002.

[22] P. Company, M. Contero, J. Otey, J. D. Camba, M.-J. Agost, and D. Pérez-López, "Web-Based System for Adaptable Rubrics: Case Study on CAD Assessment," *Journal of Educational Technology & Society*, vol. 20, no. 3, pp. 24–41, 2017.

[23] M.-J. Agost, P. Company, M. Contero, and J. D. Camba, "CAD training for digital product quality: a formative approach with computer-based adaptable resources for self-assessment," *Int J Technol Des Educ*, vol. 32, no. 2, pp. 1393–1411, Apr. 2022, doi: 10.1007/s10798-020-09651-5.

[24] J. Leith and H. K. Ault, “Using Think-Aloud Exercises to Reveal Students’ Solid Modeling Strategies,” p. 6, 2013.

[25] T. Branoff and N. Hartman, “Defining Expertise In The Use Of Constraint Based Cad Tools By Examining Practicing Professionals,” in *2004 Annual Conference Proceedings*, Salt Lake City, Utah: ASEE Conferences, Jun. 2004, p. 9.369.1-9.369.13. doi: 10.18260/1-2--13970.

[26] A. Rynne and W. Gaughran, “Cognitive modeling strategies for optimum design intent in parametric modeling (PM),” *Computers in Education Journal*, 2007.

[27] M. F. Daud, J. M. Taib, and R. S. Shariffudin, “Assessing Mechanical Engineering Undergraduates’ Conceptual Knowledge in Three Dimensional Computer Aided Design (3D CAD),” *Procedia - Social and Behavioral Sciences*, vol. 56, pp. 1–11, Oct. 2012, doi: 10.1016/j.sbspro.2012.09.626.

[28] M. Dinar, J. D. Summers, J. Shah, and Y.-S. Park, “Evaluation of Empirical Design Studies and Metrics,” in *Experimental Design Research: Approaches, Perspectives, Applications*, P. Cash, T. Stanković, and M. Štorga, Eds., Cham: Springer International Publishing, 2016, pp. 13–39. doi: 10.1007/978-3-319-33781-4_2.

[29] J. Y.-H. Chen, “Development of a Novel Computer-Aided Design Experiment Protocol for Studying Designer Behaviours,” M.A.S., University of Toronto (Canada), Canada, 2021. Accessed: Oct. 11, 2022. [Online]. Available: https://www.proquest.com/docview/2609411741/abstract/DFD0E72F58A64295PQ/1

[30] H. Ault, L. Bu, and K. Liu, “Solid Modeling Strategies – Analyzing Student Choices,” in *2014 ASEE Annual Conference & Exposition Proceedings*, Indianapolis, Indiana: ASEE Conferences, Jun. 2014, p. 24.1093.1-24.1093.15. doi: 10.18260/1-2--23026.

[31] C. Xie, Z. Zhang, S. Nourian, A. Pallant, and E. Hazzard, “Time Series Analysis Method for Assessing Engineering Design Processes Using a CAD Tool,” *International Journal of Engineering Education*, vol. 30, pp. 218–230, Jan. 2014.

[32] S. Bhavnani, J. Garrett, and J. Shaw, “Leading Indicators of CAD Experience,” *CAAD Futures ?93 [Conference Proceedings / ISBN 0-444-89922-7] (Pittsburgh / USA), 1993, pp. 313-334*, Jul. 2008.

[33] “DELPHI PROCESS: A METHODOLOGY USED FOR THE ELICITATION OF OPINIONS OF EXPERTS.” Accessed: Apr. 06, 2023. [Online]. Available: https://apps.dtic.mil/sti/citations/AD0675981

[34] R. F. Hamade and H. A. Artail, “A study of the influence of technical attributes of beginner CAD users on their performance,” *Computer-Aided Design*, vol. 40, no. 2, pp. 262–272, Feb. 2008, doi: 10.1016/j.cad.2007.11.001.

Appendix A: Case-Study Model Drawings

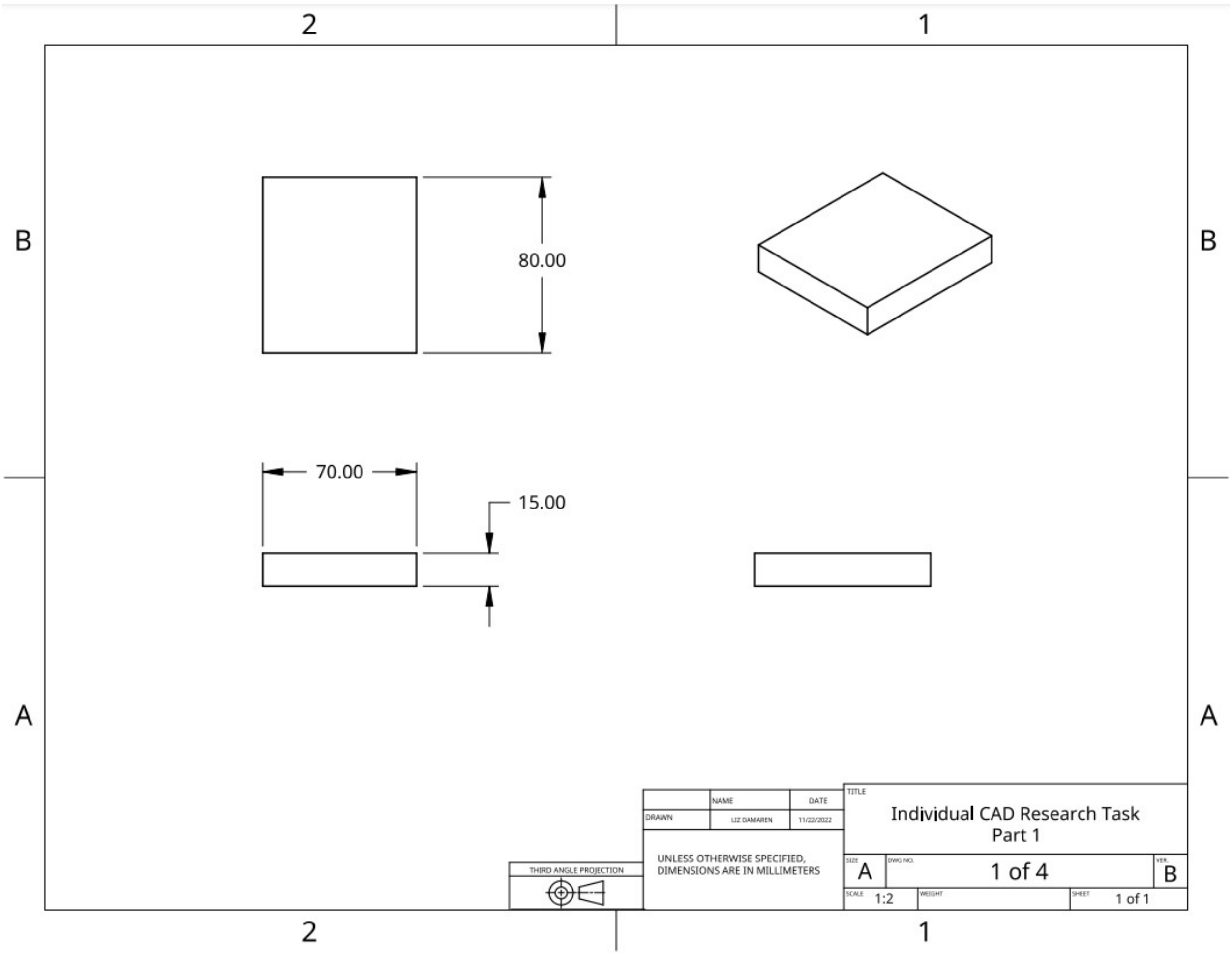

Figure 14: Phase 1 CAD Activity

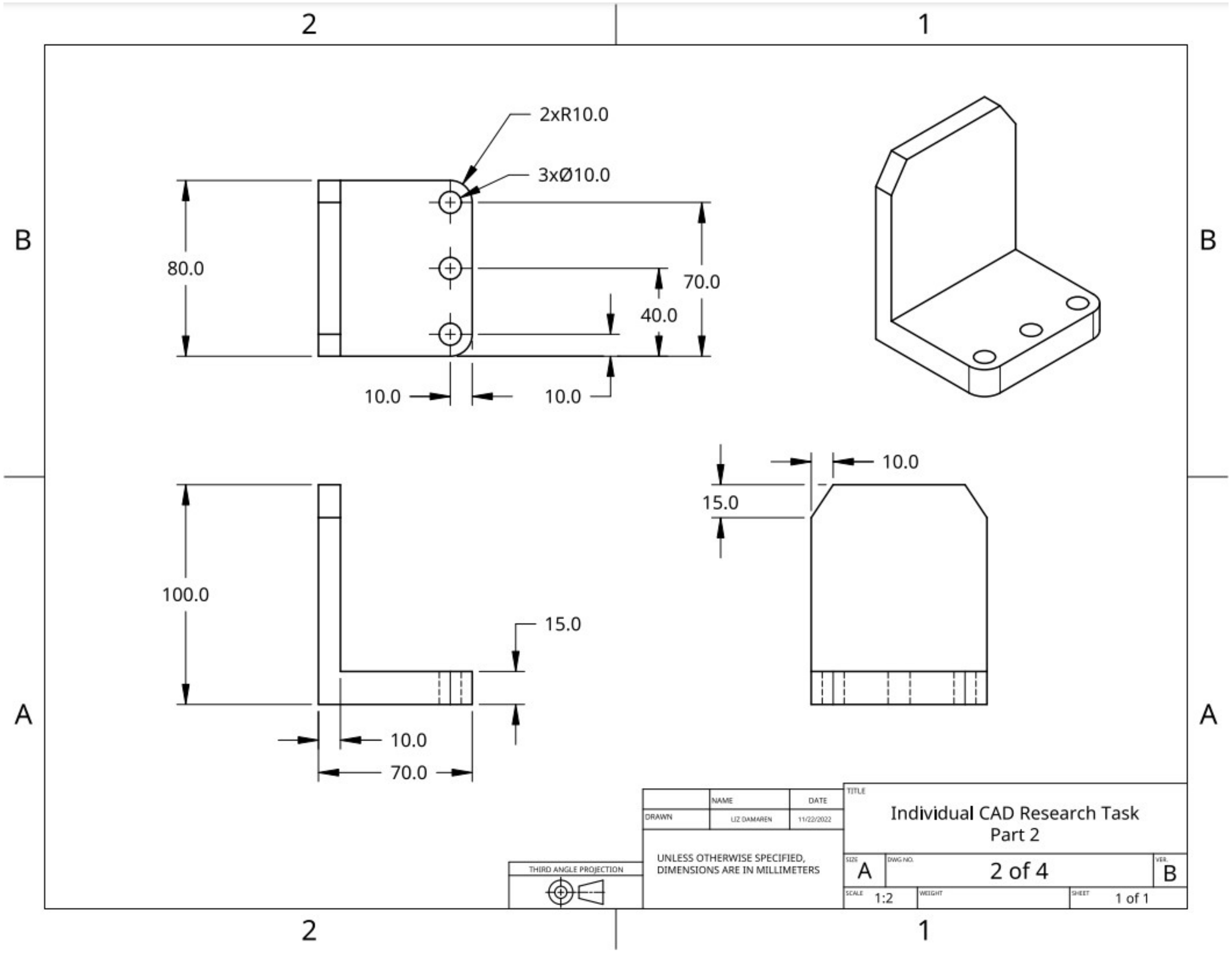

Figure 15: Phase 2 CAD Activity

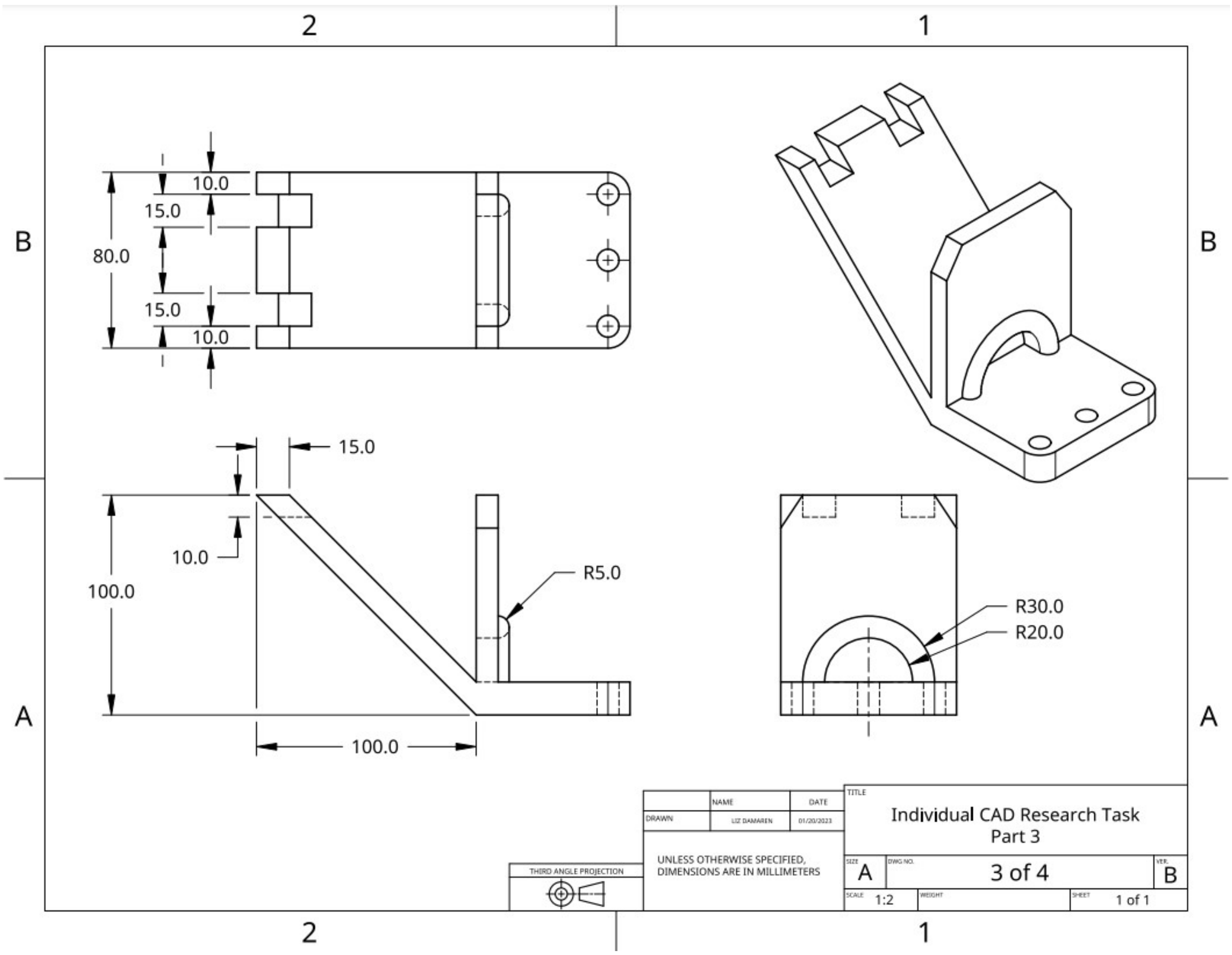

Figure 16: Phase 3 CAD Activity

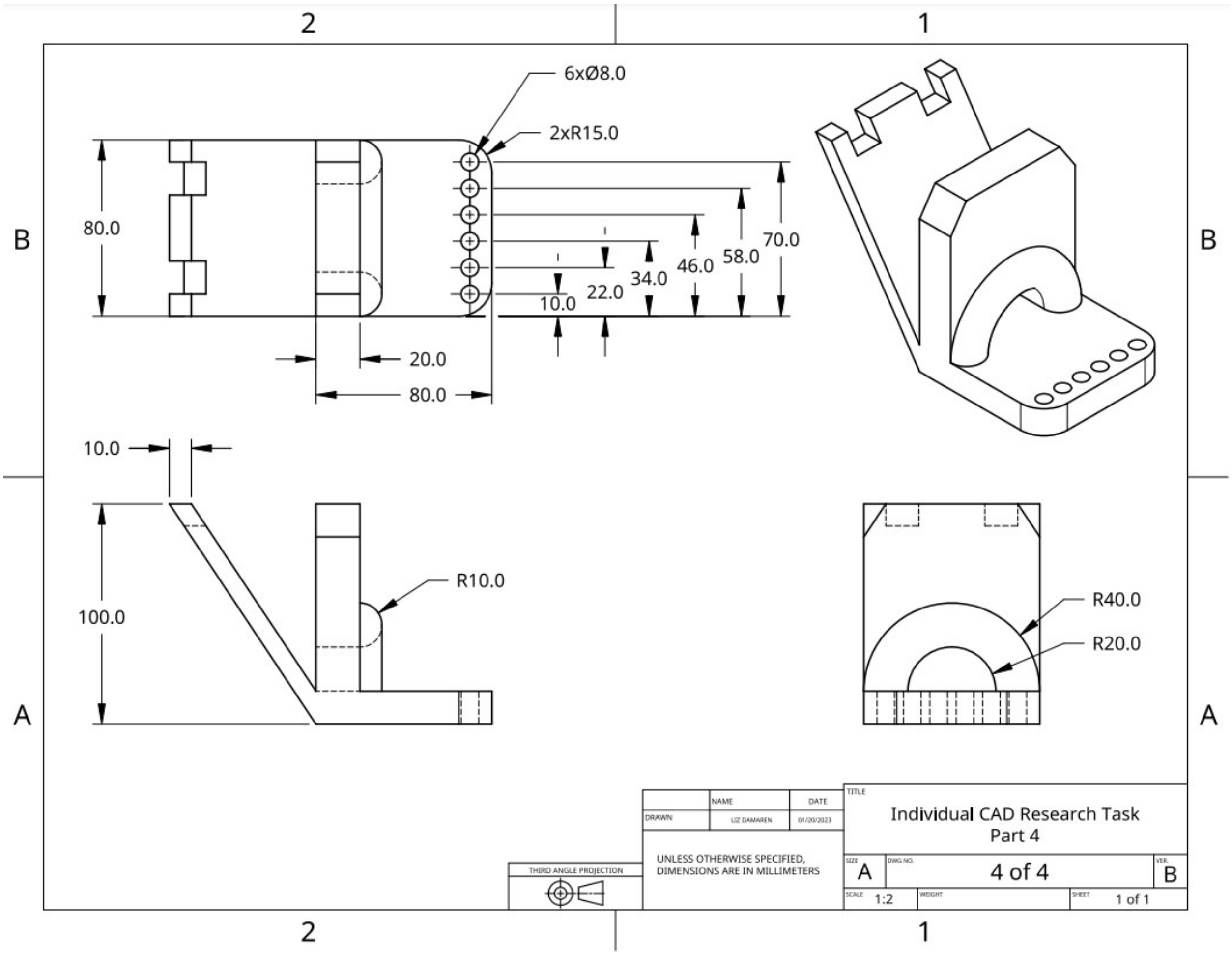

Figure 17: Phase 4 CAD Activity