

USING RAPID APPLICATION DEVELOPMENT FOR SOFTWARE DEVELOPMENT PROJECTS

by

Suyash Agrawal

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the Degree of

Master of Science



Department of Computer and Information Technology

West Lafayette, Indiana

May 2019

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Prof. Alka Harriger, Chair

Department of Computer and Information Technology

Dr. John Springer

Department of Computer and Information Technology

Prof. Kevin Dittman

Department of Computer and Information Technology

Dr. Paul Parsons

Department of Computer and Graphics Technology

Approved by:

Dr. Eric T. Matson

Head of the Graduate Program

ACKNOWLEDGMENTS

I wish to gratefully acknowledge my thesis committee for their insightful comments and guidance and my family for their support and encouragement.

TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	viii
LIST OF ABBREVIATIONS	x
GLOSSARY	xi
ABSTRACT	xiii
CHAPTER 1. INTRODUCTION	1
1.1 Statement of Purpose	1
1.2 Significance	1
1.3 Research Question	2
1.4 Assumptions	3
1.5 Limitations	3
1.6 Delimitations	4
1.7 Summary	4
CHAPTER 2. REVIEW OF LITERATURE	5
2.1 Relevant Previous Work	5
2.2 Rapid Application Development (RAD)	6
2.3 Advantages and Disadvantages of RAD	9
2.4 Usability of Software Tools	11
2.5 Conducting Surveys as an Investigation Tool	12
2.5.1 Survey Instruments	14
2.6 Using Autoethnography as a first person research Technique	15
2.6.1 Thematic analysis	16
2.7 Conclusion	18
CHAPTER 3. FRAMEWORK AND METHODOLOGY	19
3.1 Study Design	19
3.2 Autoethnography of the Application Development Experience	20
3.2.1 Prototype application	20
3.2.2 Developing the Mendix application	20

3.2.3	Autoethnography	21
3.2.3.1	Coding of the data	22
3.3	Expert Survey	23
3.3.1	Data Collection	24
3.3.2	Recruitment Strategy and Protocol	26
3.3.3	IRB Protocol	26
3.4	Data Analysis	27
3.5	Summary	27
CHAPTER 4. IMPLEMENTATION		28
4.1	Judging app development	28
4.1.1	Module development	28
4.1.2	Task tracking	29
4.1.3	Deployment	30
4.1.4	Look and Feel	31
4.2	Usability survey	32
4.2.1	Qualtrics	33
4.2.2	Mendix forum posting	34
4.3	Summary	35
CHAPTER 5. RESULTS		36
5.1	Thematic analysis of autoethnographic data	36
5.1.1	Analysis using Usability priori codes	36
5.1.2	Analysis based on emotions	38
5.1.3	Analysis for potential roadblocks	40
5.2	Survey Data	41
5.2.1	Demographic Profile	41
5.2.2	Ease of Learning	45
5.2.3	Ease of use	46
5.2.4	Usefulness	46
5.2.5	Satisfaction	48
5.2.6	Comparative survey statistics	49

5.2.7	Positive Aspects of Mendix	50
5.2.8	Negative Aspects of Mendix	50
5.3	Summary	51
CHAPTER 6. DISCUSSION AND CONCLUSION		52
6.1	Discussion of the Application development experience	52
6.1.1	Potential challenges on using Mendix	54
6.1.2	Suggestions/ Recommendations on developing applications using Mendix	55
6.2	Discussion of the survey data	57
6.3	The power of the journey - self realization	58
6.4	Future scope	59
REFERENCES		61
APPENDIX A. AUTO ETHNOGRAPHIC NOTES		67
APPENDIX B. APPROVED IRB PROTOCOL		79
APPENDIX C. RESPONSES TO OPEN-ENDED SURVEY QUESTIONS		80
APPENDIX D. OTHER AUTOETHNOGRAPHIC DATA		84
APPENDIX E. APPLICATION SCREENSHOTS		87

LIST OF TABLES

2.1	Survey Instruments	14
3.1	Coded task themes	23
3.2	Coded emotions	23
3.3	Generating themes using inductive coding	24
4.1	Survey Questions	32
5.1	Thematic analysis based on usability	37
5.2	Age group distribution	42
5.3	Educational background distribution	43
5.4	Ease of Learning	45
5.5	Ease of use	46
5.6	Usefulness	46
5.7	Satisfaction	48
5.8	Comparison of survey data based on educational background	49
5.9	Comparison of survey data based on experience	49

LIST OF FIGURES

3.1	Thesis Timetable	19
3.2	USE Questionnaire	25
4.1	Story dashboard with list of tasks and their status	29
4.2	Mendix commit history	30
4.3	Mobile view of new application	31
4.4	Usability survey preview	33
4.5	Mendix community forum post	34
5.1	Thematic analysis based on usability	36
5.2	Thematic analysis based on emotions	38
5.3	Thematic analysis on points of frustration	40
5.4	Gender distribution	41
5.5	Educational background	42
5.6	Current employment status	43
5.7	Frequency of usage	44
5.8	Duration of usage	44
5.9	Ease of Learning	45
5.10	Ease of use	47
5.11	Usefulness	47
5.12	Satisfaction	48
5.13	Word cloud for positive responses	50
5.14	Word cloud for negative responses	51
6.1	Microflow as viewed in Mendix	52
6.2	Domain model as viewed in Mendix	53
B.1	IRB exempt status	79
E.1	Old login screen	87
E.2	New login screen	87
E.3	Old admin dashboard	88
E.4	New admin dashboard	88

E.5	Old score management page	89
E.6	New score management page	89
E.7	Old user management page	90
E.8	New user management page	90
E.9	Old group management page	91
E.10	New group management page	91
E.11	Old rubric management page	92
E.12	New rubric management page	92
E.13	Old topic per rubric page	93
E.14	New topic per rubric page	93
E.15	Old questions per topic page	94
E.16	New questions per topic page	94
E.17	Old judge landing page	95
E.18	New judge landing page	95
E.19	Old group selection page	96
E.20	New group selection page	96
E.21	Old questionnaire page	97
E.22	New questionnaire page	97

LIST OF ABBREVIATIONS

abbr	abbreviation
AJAX	Asynchronous JavaScript And XML
CGT	Computer Graphics Technology
CIT	Computer and Information Technology
CSS	Cascading Style Sheets
GUI	Graphical User Interface
HTML	Hypertext Markup Language
ISDM	Information System Development Method
IT	Information Technology
RAD	Rapid Application Development
RIA	Rich Intent Application
SaaS	Software as a Service
TECHFIT	Teaching Engineering Concepts to Harness Future Innovators and Technologists
XML	Extensible Markup Language

GLOSSARY

Asynchronous JavaScript And XML – A set of Web development techniques using many web technologies on the client side to create asynchronous Web applications.

Autoethnography – A form or method of research that involves self-observation and reflexive investigation in the context of first person research field work and writing (Maréchal, 2010).

Cascading Style Sheets – A style sheet language used for describing the appearance of a document written in a markup language like HTML.

Extensible Markup Language (XML) – A markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

Graphical User Interface – A form of user interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, instead of text-based user interfaces, typed command labels or text navigation.

Hyper Text Markup Language – The standard markup language for creating web pages and web applications.

Judging App – An application that provides a method to evaluate competitions based on predefined criteria.

Rapid Application Development – An integrated set of techniques, guidelines and tools that facilitate deploying customer's software needs within a short period of time (Gottesdiener, 1995).

Rich Internet Application – A Web application that has many of the characteristics of desktop application software, typically delivered by way of a site-specific browser, a browser plug-in, an independent sandbox, extensive use of JavaScript, or a virtual machine.

Software as a Service – a software licensing and delivery model in which software is licensed on a subscription basis and is centrally hosted.

Teaching Engineering Concepts to Harness Future Innovators and Technologists – An NSF-funded project that seeks to spark interest in middle school students in STEM subjects by showing them that these skills can equip them with the tools to innovate solutions to societal problems such as obesity by creating their own exergames (Harriger, Harriger, Flynn, & Flynn, 2015).

Usability – An approach to the design of technological interfaces which attempts to make them intuitive and easy to use (Chandler & Munday, 2011).

ABSTRACT

Author: Agrawal, Suyash. M.S.

Institution: Purdue University

Degree Received: May 2019

Title: Using Rapid Application Development for Software Development Projects

Major Professor: Alka Harriger

In the modern era where technology is constantly evolving, it is important to evaluate new technological tools and approaches in order to gauge their potential for adoption. Rapid Application Development (RAD) has highly evolved over the years, but it has not seen much response at the university level (*University Program - Free Application Development Software Resources*, n.d.). Several studies show that perceptions of usability form the basis of acceptance or rejection of new tools and applications. Thus, running a usability study on a specific RAD tool coupled with autoethnographic documentation of specific development experience with that tool has the potential to encourage university faculty/staff to consider teaching/using it.

This research study seeks to understand developers' perceptions regarding the usability aspect of Mendix, a Rapid Application Development (RAD) tool. Both qualitative and quantitative approaches were employed to discover and understand the extent of perceived usefulness, ease of use and satisfaction with the tool. The results of the study presented a case for academicians on the viability of teaching RAD or using Mendix for their web application development needs. The study ultimately sought to help university faculty understand what to expect while teaching RAD to students from computing background and also help them decide if they would like to use tools like Mendix for their software development projects or continue to use specific coding languages (traditional software development) for software development projects.

CHAPTER 1. INTRODUCTION

This chapter provides an overview of the research study. It covers the research significance, assumptions, limitations and delimitations which define the extent of the study.

1.1 Statement of Purpose

The purpose of this study was to understand the perceptions of usability when using RAD tools, specifically Mendix, for web development projects. The study examined Mendix from a usability perspective by illustrating the researcher's experience of developing a specific web application for a university project with Mendix and evaluating results from a usability survey of experienced Mendix users.

First, Mendix, which is the most popular RAD platform (Marvin, 2018), was used to build a judging web application for a university project called Teaching Engineering Concepts to Harness Future Innovators and Technologists (TECHFIT) (Harriger et al., 2015). Additionally, a usability survey was conducted to identify the perceived usability of Mendix by developers in industry (referred as experts in this study). The observations from the researcher's detailed autoethnographic account along with the usability survey results from the experts can aid academicians/project leaders who are considering to teach RAD or use RAD tools for future software development projects.

1.2 Significance

Though RAD has been around for a few decades and ever-evolving technology demands more and more applications to be built, there has been very little acceptance of RAD as a potential application development tool by universities (*University Program - Free Application Development Software Resources*, n.d.). Most of these universities still prefer teaching traditional application development and require new software to be built using a specific coding language. While Mendix offers a University program for

professors to empower their students to build web and mobile applications at no cost, only 130 professors across 81 universities have accepted Mendix's offer to use it (*University Program - Free Application Development Software Resources*, n.d.) which is a very small fraction of 1 million+ professors across 4000+ schools across the United States. The lack of recent academic literature on usability of RAD tools added to the need for a usability study to be conducted to gain insight on the perceptions of satisfaction, ease of use and usefulness of Mendix.

The combination of autoethnography for a specific development experience by a university graduate student along with the usability survey may be able to provide a better understanding of the perceived benefits or deterrents of the tool.

The results of the study may be able to present a case for academicians to use Mendix for teaching software development or for completing or implementing or similar software project needs. The study ultimately aims to help university faculty understand what to expect while teaching RAD to students with computing background and also help them decide if they would like to use tools like Mendix for their software development projects.

1.3 Research Question

1. What is the experience of developing an application in Mendix for an intermediate developer from a computer science background?
 - (a) What were the major points of frustration or road blocks in the development process?
 - (b) What were the usability aspects that enhanced the development process?
 - (c) What was the emotional journey of the developer during the development process?
2. What are the perceptions of usability on Mendix among experts?

1.4 Assumptions

The following assumptions were inherent to the design of this study:

- All participants will answer the questions honestly.
- Number of responses received for the survey will be appropriate for statistical analysis.
- Proper technological infrastructure is available to facilitate the survey.
- All features of the judging application can be completed by the sole developer in the time allotted.
- The researcher will be able to document his development experience in a descriptive fashion.

1.5 Limitations

The following limitations were inherent in the design of this study:

- Participation in the survey will be voluntary, thus the number of participants cannot be controlled.
- Participants may be unable or unwilling to share or describe their experiences in response to the open-ended questions asked during surveys.
- The findings of this study cannot be generalized as the study is limited to a specific RAD tool (Mendix) and exclusive to a specific development exercise (TECHFIT Judging Application) by a single developer.
- Usability perceptions are limited to three attributes: usefulness, satisfaction and ease of use.

1.6 Delimitations

The following delimitations were inherent to the design of this study:

- The researcher will be the sole developer of the judging web application for TECHFIT.
- The researcher-developed application will be based on the ASP.Net web application currently in place on the TECHFIT website.
- The new RAD application will be built using Mendix.
- The researchers autoethnographic documentation will be based solely on the development of the judging application using Mendix.
- The development work will be done in parallel to the survey data collection to minimize influence of the survey data in the autoethnographic development study.
- The usability survey results will be solely based on responses from the Mendix community forum, which consists of experienced developers who have knowledge of using Mendix.

1.7 Summary

This chapter provided the scope, significance, research question, assumptions, limitations, delimitations, definitions, and other background information for the research project.

CHAPTER 2. REVIEW OF LITERATURE

2.1 Relevant Previous Work

The methodology for this study was inspired by Romina Laura Bot's master's thesis titled "A study of perceptions usability and future adoption of an electoral database" (Bot, 2013). Bot's research study sought to understand students' and professors' perceptions of usability of an electoral database to explain the intention of adopting the electoral database as a learning tool for political scientists. Bot explained how relevant literature showed that perceptions towards the usability of a product can lead to the decision towards adoption and ultimately usage of the product. The work included a usability study that employed a survey that was developed by comparing and combining various pre-validated survey instruments and modifying the questions to suit the study. The sample population for Bot's survey was students and professors in the area of political science at Purdue University. The survey results were then analyzed using various qualitative and quantitative methods in order to discover relationships between usability and perceptions of usage. The results of the study provided valuable insights into the impact of usability and innovation in user's perception that might contribute to the acceptance of the tool.

The proposed research study takes inspiration from Romina's work on running a usability study using a survey method for evaluating the RAD tool - Mendix. It further extends Bot's approach by documenting a personal development experience using autoethnography to provide additional insights into the usability aspect from an individual developer's perspective. The insights from the personal account will be compared and contrasted to the survey results to find commonalities and differences. The combination of the two approaches has the potential for providing a better case for academicians to potentially accept or reject the tool.

2.2 Rapid Application Development (RAD)

Rapid Application Development or RAD is a form of Agile software development methodology, which was invented by James Martin in 1991 (Martin, 1991). He wanted to respond to the limitations of the prevalent waterfall methodology for developing software. Unlike the waterfall method, RAD provides an iterative method that supports fast prototyping and enables developers to get multiple user feedback. The concept of RAD was anticipated to be better than all Software Development Life Cycle (SDLC) methods available during that time. James Martin's approach divided the process into four phases as:

1. Requirements planning phase
2. User design phase
3. Construction phase
4. Cut-over phase

The idea pioneered by James Martin was further developed and improved by RAD pioneers such as James Kerr and Richard Hunter. Kerr and Hunter (1993) brought into light the journey of a RAD project manager who implemented RAD methodology on a live RAD project. This provided further evidence for RAD practices as an alternative to the traditional development techniques prevalent at that time.

The problem with traditional software development methods is that they were designed on the basis of traditional engineering models used in other fields. The way a building is built is very different from the way software is built. It is imperative to understand that the entire process is highly dependent on the software and its requirements. In most cases, useful insights gained during the development process can be fed back into designing a better solution (Brooks & Kugler, 1987). Thus when compared to traditional development approaches, RAD provides more flexibility in order to be able to use the insights gained during the project and hence make a better solution.

Though RAD gained popularity in the early nineties, it was widely confused as an excuse for sidetracking the software engineering standards (Howard, 2002). Initially RAD failed to materialize the high hopes of providing a powerful formula for coping with project delays and increasing developer productivity. RAD solutions were suspected to decrease the product quality since they did not fit in any traditional models. Other concerns included the extra cost of user involvement and prototyping (Howard, 2002).

Many traditional software developers consider RAD as an excuse to avoid the rigorous disciplines required to build a reliable, functional system (Rapley, 1995). They observe that the need of documentation is often sidelined owing to the emphasis on fast delivery of the product. The missing documentation can have wide-scale implications later. This can result in software failures and unmaintainable components (Rapley, 1995). The assumption of traditional development practice, in this case, is that it will always have a very high level of documentation. Moreover, amongst the differences in these approaches, it is important to realize the fact that no single method will fit for all situations. Thus, the need to see which approach will suit the most for a project is important.

RAD implements an iterative development approach which compresses the software development life-cycle into short iterations such that the results can be demonstrated to various stakeholders. The system is typically built on the ideology of developing functionality in an iterative way by generating prototypes at the end of each iteration and getting early user feedback (Berger & Beynon-Davies, 2009). Thus it is a way to get ongoing support from management by showing increasing and visible progress through each new prototype.

A thorough content analysis of rapid development stems from business management literature as well as software development sources. While the former talks about the incentives of having quick time to market, the later establish newer sources of management principles (Smith, Colombi, & Wirthlin, 2013).

Initially, it was assumed that most RAD projects are suited for applications that require high levels of front-end interactions and considerably low level of back-end complexity (Beynon-Davies, Mackay, & Tudhope, 2000). The ideology ruled out RAD for large-scale complex projects with vast databases. Thus, for all real-time systems, large infrastructure systems, critical security systems etc. RAD was deemed inappropriate. Though there is still disparity amongst people for the same, the Air Force Research Laboratory (AFRL) and Department of Defense (DoD) have realized the importance of having rapid development, which can enable their teams to deliver an 80 percent solution within 6-12 months. These projects are termed as rapid development projects and are often the driving factor for new innovative solutions (Smith et al., 2013). Another such example is the Advance Technology Centre of BAE Systems that uses rapid development for supporting software-intensive embedded systems that require rapid design and deployment (Jones & Leung, 2005).

RAD has been compared with an agile Information System Development Method (ISDM). While ISDM itself is considered as a technological innovation for an organization, distinct features of RAD can be compared to be an instance of ISDM (Berger & Beynon-Davies, 2009). The modern low code development platforms are essentially modern RAD systems. These are widely essential for businesses that are undergoing transformation (Ross, 2018). The approach highly depends on the level of commitment by the managers as well as the users of the application. It relies heavily on users to give feedback in a timely manner as well as on stakeholders to take decisions based on such feedback in a time efficient manner. The inaccurate capture of the importance of having multiple iterations can lead to an incorrect evaluation for project managers. Therefore, at least three iterations are recommended (Smith et al., 2013).

In newer Information Technology (IT) environments, many systems use RAD to a certain extent (Hotle, 2014). In the modern world, which is mostly app-centric, nearly everything uses the Software as a Service(SaaS) model. The demand of applications is rapidly increasing, and the low code development platforms can be used to deploy such applications in a small time-frame (Marvin, 2018). Such platforms can provide a working application that can be downloaded and used within a few days, and the functionality can be enhanced using iterative development.

Rich Internet Applications (RIAs), which provide the user with a responsive and interactive experience with their well-designed highly responsive Graphical User Interfaces(GUIs), are gaining popularity (Dissanayake & Dias, 2014). These are very different from the traditional, slow web pages that have limited responsiveness and interaction. With the advent of HyperText Markup Language 5 (HTML5) and Cascading Style Sheets 3(CSS3), demand for RIAs has grown. Another important innovation in this respect is Asynchronous JavaScript And XML (AJAX), which supports RIAs. RAD combines these technologies and provides a reliable formula to build RIAs that have top-notch quality and productivity.

2.3 Advantages and Disadvantages of RAD

The advantages of RAD include:

1. Better quality: As the users get to interact with the prototypes, the resulting product has higher quality and a higher user acceptance rate. A higher level of business functionality is achieved, which ensures better addressing of the business problem (Beck & Gamma, 2000).
2. Better risk control: Though the literature has not explored the risk factors associated in project development, the RAD approach actually mitigates risks by focusing on key business objectives, and the iterations ensure that those risks are addressed in a timely and effective manner (Beck & Gamma, 2000).

3. Better project on-time completion: Using an iterative development technique combined with constant user involvement decreases the chances of project failures to a great extent. Issues are identified early in the process and changes can be made accordingly in future iterations that reduce the chances of project delays and failures (Beck & Gamma, 2000).

In order to provide a balanced overview of the literature, it is important to look into the disadvantages of RAD as well. Some of these might not be completely relevant given the sources of the cons are dated but it will give a holistic view of the RAD domain. The disadvantages include (Begel & Nagappan, 2007; Maximilien & Williams, 2003; Rosenberg & Stephens, 2008)

1. Relatively new approach: As humans are averse to changes, accepting a new methodology for software development has a lot of resistance. The fear of failure of implementing a new approach for the first time acts as a disadvantage to RAD. As traditional software development started in early 1960s and RAD came up in early 1990s and has started gaining popularity only a few years back, it is relatively new.
2. Steep Learning Curve: In relation to the point above, a new approach would require the developers to learn a new methodology and implement it. This requires additional efforts.
3. Lack of expertise: New technology brings in the challenge of lack of subject matter experts in that field. In the case of RAD, people with experience in traditional software development will have to start learning the new method from scratch and build their skills with time.
4. Constant user interaction: Unlike traditional development, RAD requires the users of the system to be constantly involved with the system development process. Their feedback is important for each iteration, and getting time from the users can be tough at times.

5. Lack of scalability: As mentioned above in the literature, RAD has been focused on small and medium scale projects. Moreover, the lack of complete control over the design might affect the projects adversely in case of large-scale projects (Begel & Nagappan, 2007; Maximilien & Williams, 2003; Rosenberg & Stephens, 2008).

Establishing the metrics for evaluation of such low code application development platforms is highly recommended. Though the tools such as Zoho, Salesforce App Cloud, Mendix, Appian etc. perform similarly, they differ in the ease of use, breadth of functionality and overall efficiency (Marvin, 2018). Thus, evaluation between traditional software development techniques and RAD needs to be done on the basis of its integration with business, the ease of embracing new technology, and the effectiveness of pre-built components (Suri, Kumar, & Singh, 2011).

2.4 Usability of Software Tools

Brooke et al. (1996) defines usability as not a quality that exists in any real or absolute sense. It is a general quality of the appropriateness to a purpose of any particular artifact. Thus, usability for any interface needs to be evaluated with reference to the context in which the interface will be used. The appropriateness of the usability test can then be decided accordingly (Bevan, 1991). Moreover, in order to correctly decide the usability test, the following should be defined:

1. Intended users of the system
2. Tasks that the users will perform
3. Definition of environment in which the system will be used (Zazelenchuk & Boling, 2003)

Usability issues are evaluated in usability tests. In a typical scenario, the list of potential issues would include the greatest problems for real-world users. One approach to prioritize these usability issues might be to rank them on the basis of severity, but studies show that the overestimation and underestimation of severity can be an issue when performing such a study (Affairs, 2013).

In order to understand the contextual environment, which will include a web-based portal application, the user experience may vary on the basis of previous web application user experience, type of web applications used, etc. The user satisfaction of the portal will not just be based on the overall design of the system but also on these factors (Xiao & Dasgupta, 2005).

Proposed solutions or design types can be evaluated on the basis of questionnaires that can be completed prior to and following the experiment. Usability questionnaires to obtain end-users' opinions about the system are standardized and can be used as a recommended investigation tool in combination with other usability techniques (Albert & Dixon, 2003).

2.5 Conducting Surveys as an Investigation Tool

The basic idea of survey methodology is to collect information from a group of people by sampling individuals from a large population. However, when it comes to software related surveys, there are some visible differences, such as the people taking the survey are generally employed in software companies and the questions are also specific to software engineering (LinÅker, Sulaman, de Mello, Hst, & Runeson, 2015). Because the methodology is based on sampling, a fixed-design approach is followed under which research is categorized into the planning phase and the execution phase.

Firstly, the research objectives need to be defined. These help in identifying the interest and providing a guideline in terms of scope and context, which ultimately confines the researcher from steering away in wrong directions.

The purpose of research can be descriptive, explanatory, exploratory. While descriptive research supports making claims or assertions about the population in general, the explanatory survey helps the researcher explain trends or problems observed in the population. The exploratory research is generally used to get new insights into an unknown area (Wohlin et al., 2012).

The next step is to identify the target audience. The target audience consists of the people who can provide relevant information in order to achieve the research objective. Kasunic (2005) presents the following set of demographical attributes that are made specifically for the target audience in software engineering surveys:

- size (I)
- jobs and responsibilities (I)
- education level (I)
- gender (I)
- age (I)
- technical abilities (D)
- relevant experience (D)
- perception regarding the survey' domain knowledge (D)

where "D" represents the dependent and "I" represents the independent attributes respectively.

The next aspect is choosing the survey instrument. The researcher can decide to use a pre-validated survey instrument or design a new survey instrument for their study. The decision is based primarily on the amount of time and resources available for the study.

The process begins with designing the questionnaire and deciding the questionnaire type. Two of the most popular survey types are self-administered and interviewer-administered questionnaire. Web-based questionnaires are categorized under self-administered survey methods and are generally easy to set up and distribute. They preserve confidentiality and remove any interference or influence from the researcher (LinÅker et al., 2015).

Some popular web-based questionnaire websites are SurveyMonkey, QuestionPro, Qualtrics, Google forms etc.

2.5.1 Survey Instruments

Various questionnaires have been used to evaluate user interfaces and usability of systems (Root & Draper, 1983). Nine questionnaires designed specifically to assess aspects of usability are as follows (Schneider, 2018):

Table 2.1. Survey Instruments

Acronym	Instrument	Institution	Contents
QUIS	Questionnaire for User Interface Satisfaction	Maryland	27 questions
PUEU	Perceived Usefulness and Ease of Use	IBM	12 questions
NAU	Nielsen's Attributes of Usability	Bellcore	5 attributes
NHE	Nielsen's Heuristic Evaluation	Bellcore	10 heuristics
CSUQ	Computer System Usability Questionnaire	IBM	19 questions
ASQ	After Scenario Questionnaire	IBM	3 questions
PHUE	Practical Heuristics for Usability Evaluation	OSU	13 heuristics
PUTQ	Purdue Usability Testing Questionnaire	Purdue	100 questions
USE	USE Questionnaire	Sapient	30 questions

2.6 Using Autoethnography as a first person research Technique

As technological enhancements increase the integration of technology with our daily activities, considerable research has been done on analyzing the human-computer interaction (HCI) realm (Desjardins & Ball, 2018). The first person research is a viable addition to traditional HCI methods. It helps in collecting and analyzing data from experiences of researchers themselves rather than focussing on external users (Lucero et al., 2019). Autoethnography and autobiography are the more common approaches for conducting first-person research (Cecchinato, Cox, & Bird, 2017; LeCompte, Schensul, et al., 2012; Lucero et al., 2019).

Autoethnography initially originated from qualitative research in social sciences in the early 1980s (Ellis, Adams, & Bochner, 2011). Though there have been concerns about the subjectivity of the account and personal perspectives, Ellis and Bochner emphasize how autoethnography supports in shaping the research by accounting for the subjectivity. Autoethnography as a reflexive method to express and evaluate the researcher's lived experience from the researcher's perspective is reported as stories narrated in first person.

Though running an autoethnography study and evaluating it is not a straight forward task, there are some basic guidelines which should be followed during the process. Medford (2006) suggests that a primary ethical standard must be followed in writing the personal account and the evaluation as well. Megford further suggests that the writer should be willing to explain his/her experience and confront any issues that might arise due to any disagreement on the representation of any shared experiences of the subjects. Furthermore, Ellis and Bochner (2000) states that a good autoethnographic narrative should be able to express the feelings and thinking of the person along with the experience, position of author, learnings and description of an event.

Autoethnography projects have become more prominent during recent years where more and more studies are getting published such as non-routine usage of mobile devices (Lucero, 2018), personal fitness and self-tracking, black men in the IT workforce, the experience of skateboarding, smartwatches (Cecchinato et al., 2017), personal heritage soundscapes, horseback riding to design (Höök, 2010) etc. Studies support the concept of

autoethnography as a first-person research technique that offers a varied epistemological view to help researchers investigate lived experiences and gain deep insights (Desjardins & Ball, 2018). This personal inside perspective is uncommon among other research techniques. Thus, autoethnography can provide a deep understanding of personal experience that can lend support in the evaluation of a technology or a tool.

2.6.1 Thematic analysis

One of the qualitative research methods available for conducting analysis of the data, thematic analysis (TA) is the process of identifying themes or patterns in qualitative data. TA is poorly branded yet widely used in qualitative research (Braun & Clarke, 2006). It is often referred by beginners in qualitative research as it provides the basic skill set that can be used to conduct analysis. It is a flexible method that can be used in different ways. In various instances Braun and Clarke (2006)'s six step framework has been referred for running thematic analysis on data. The steps are as follows:

1. Familiarize with data

The first step in the qualitative analysis which aims at getting familiarized with the data by reading and re-reading the transcripts.

2. Generate initial codes

In this phase, the data is organized in a meaningful and systematic way. It reduces large chunks of data in to small word codes that represent that data.

3. Search for themes

The codes are searched for any particular pattern or something that captures significant or interesting aspect related to the research question. There are no set rules on what comprises of a theme (Braun & Clarke, 2006), thus a theme is categorized based on significance.

4. Review themes

The themes developed in previous step are reviewed, modified and tested for relevance. Data categorized in a particular theme is checked for direct representation or support of the theme.

5. Define themes

This step aims in defining the themes and any sub themes that are evident from the data analysis. Efforts are made to find any correlation or interactions between the themes.

6. Write up

The end part of the analysis is generating a report that can be included as part of the document with the broad analysis, learnings and interpretations that emerge from the thematic analysis.

In addition to the basic step by step process of running thematic analysis, there are various ways of approaching thematic analysis such as:

- Inductive: coding and theming based on content of the data.
- Deductive: coding and theming by existing concepts or ideas.
- Semantic: coding and theming reflect explicit content of data.
- Latent: coding and theming focused on underpinning the data.
- Realist: focused on reporting an assumed reality evident in the data.
- Constructionist: focused on how a reality is created by the data.

2.7 Conclusion

Though Rapid Application Development (RAD) has been around for a few decades, limited research has been done to probe the RAD tools from a usability standpoint. Literature suggests that RAD supports the development of better and faster applications, but RAD has not had success in terms of adoption by universities to support their day-to-day projects, preferring the traditional approach of development of software applications.

The literature review also showed how usability is an important aspect in determining the perceptions towards adoption of a tool. Further research in the literature shows how different, pre-validated usability survey instruments can be used to get insights on the perceived usefulness and usability of a tool. The literature review also showed how autoethnographic and autobiographical approaches can offer deeper insights from a personal perspective to aid in decision-making for future users or researchers.

This study seeks to understand the perceptions of industry experts on a popular RAD tool - Mendix from a usability standpoint by conducting a usability survey. The analysis of the survey combined with the autoethnographical account of the researcher on a single development project aims to evaluate the potential of adoption of Mendix as an application development tool for university research projects. Additionally, the researcher will employ autoethnography to gain additional usability insights by building a judging application for TECHFIT using the Mendix RAD tool and comparing his development experience with the expert feedback from the surveys. The commonalities and differences from a usability standpoint may provide useful information when considering development alternatives for future software projects.

This chapter provided a review of the literature relevant to the topic of research. The next chapter provides the framework and methodology proposed for the research project.

CHAPTER 3. FRAMEWORK AND METHODOLOGY

This chapter presents the methodology followed in order to answer the research questions. It outlines the overall study design and discusses the individual components of the study in detail including the survey instrument used, data collection method and application development specifics.

3.1 Study Design

The purpose of this study was to explore the perceptions of usability in using Mendix as a development tool for applications. The study was conducted in three parts. The first part consisted of a case study that employed autoethnography to document the process of developing the TECHFIT judging application using Mendix. The second part consisted of sending pre-validated usability surveys and collecting feedback from experienced Mendix developers (referred to as experts in this study). The third part involved comparing the feedback from experts against the experience of the researcher to find commonalities and differences from a usability perspective. Figure 3.1 provides a timetable of key steps that make up this study.

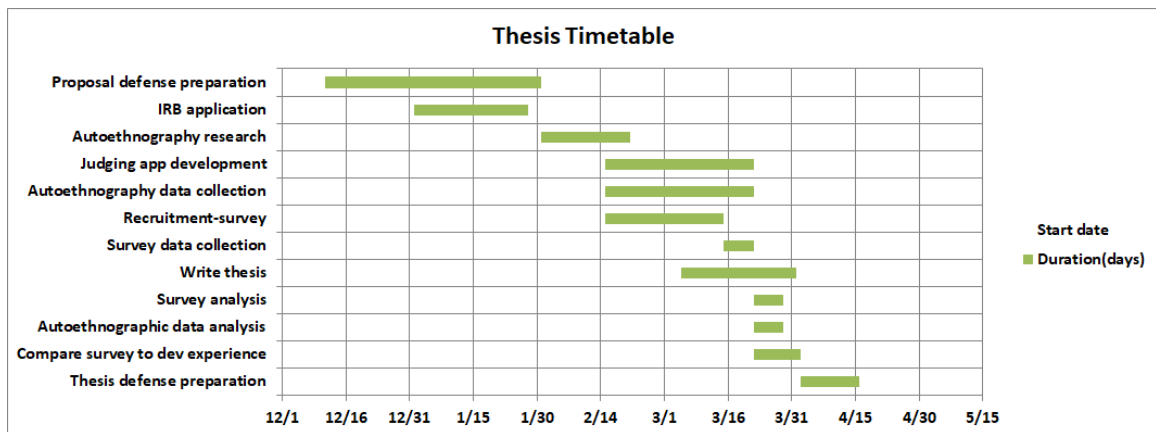


Figure 3.1. Thesis Timetable

3.2 Autoethnography of the Application Development Experience

The first part of this study concentrated on building the TECHFIT judging application using Mendix. The researcher used the current judging application as a reference point and guideline to build the new system. Autoethnography was used to capture the emotions, feelings, and the process that the researcher experienced while developing the application.

3.2.1 Prototype application

The current judging application was built using C.Net. It facilitates the judging process in TECHFIT competitions (such as Video Showcase, National Showcase competition, etc.) by providing a method to evaluate competitions based on predefined criteria. The application supports two user roles: administrator and judge. The evaluation parameters for a given event are preset by the administrator using the tool. The groups to be evaluated using specific, predefined criteria are also preset by the administrator using the tool. Judges use the tool to evaluate the assigned groups with the set criteria and are able to revise at will until the final submission of each group's evaluation. Judges may view their own overall rankings of their assigned groups. The administrator is able to view the combined rankings using all judges' evaluations for a given event.

3.2.2 Developing the Mendix application

The new judging application was built by the researcher using one of the most popular RAD frameworks: Mendix. The new application provides the same functionality that the current application provides. The process of building the application was organized into different modules to manage different aspects of the system:

1. Managing judges' accounts
2. Managing groups to be evaluated for a given event

3. Creating evaluation criteria for a given event
4. Viewing judges' evaluations for a given event

3.2.3 Autoethnography

As mentioned earlier, autoethnography was used as a method to express and evaluate the researcher's lived experiences in order to get the researcher's perspective on the usability of Mendix. For the scope of this research the following objectives and methods were followed:

Objectives:

1. To identify how a developer interacts with the product in his/her daily life.
2. To discover potential decision points during the process.
3. To find the perceived benefits of the product from a usability standpoint.
4. To uncover the issues and problems faced during the development process.
5. To understand the emotional journey that the developer goes through during the process.

Methods:

1. The researcher took regular notes on the development process.
2. The researcher recorded the emotion and task in hand before starting each development session.
3. The researcher recorded a summary of the work accomplished towards the end of the development session.
4. The researcher made additional notes whenever there was a change in emotion which resulted by an accomplishment of a task or inability to solve an error.

5. Audio recordings were converted to written notes using speech recognition on Siri and Google docs. In order to account for the difference in spoken English accent and other misinterpretations, the researcher manually reviewed and corrected each transcript.

The researcher took speech-to-text notes at various phases of the development. While majority of the notes were made before starting the development session or after completing the session, some additional notes were taken in between the process to note realizations, observations and change in feelings during the process. The data was then cleaned to remove irrelevant pauses and verbal fillers. The raw data presented in Appendix A represents the complete set of notes taken during the development process which was spread across two phases.

In addition to the methods mentioned above, the researcher also conducted semi-structured interview (personal reflection) after each phase of development. The interviewer was given a brief overview about the project and a basic guideline before starting the interview. The interviewer selected in this case was an undergrad student intern who works on Mendix as part of his part-time job. More details on the personal reflections are available in Appendix D.

3.2.3.1 Coding of the data

In order to understand the data better, the first iteration of coding consisted of running word frequency analysis on the basis of tasks and emotions. Refer to Appendix C for more details on Table 3.1, Table 3.2.

Following the guidelines from the literature, the next iteration of coding the data was carried out by assigning two to four word codes which represent each observation. Table 3.3 shows the major themes that were generated from the inductive coding. Refer to Appendix A for more details.

Table 3.1. Coded task themes

Theme	Words	Frequency
Task Management	tasks, stories, sprint, feedback	25
Business logic	domain, microflow, enumeration, relationship	20
UI Design	CSS, look and feel	9
Page Development	page, dashboard, button, grid	73
User management	permission, role	12
Deployment	commit	16

Table 3.2. Coded emotions

Word Groups	Frequency
confident, decisive	7
thinking, processing, productive, ideation, thoughtful, researching, working	10
confused, puzzled, uncertainty, clueless, oops, question	11
positive, happy, very happy, exited	18
accomplished, satisfied, know what to do	10
tired, sad, frustration, irritated, troubled	7
relieved, relaxed	3
easy, fun, amused	3
neutral	8
Misc: waiting, ready, searching, bored, trying	7

3.3 Expert Survey

The second part of this study concentrated on getting feedback from expert Mendix developers and evaluating their answers to gain insights about the perceived benefits and challenges from a usability perspective. This survey was conducted in parallel with the development of the application. Qualtrics was used as the survey tool as it was available free of cost and was easy to use to create the survey.

Table 3.3. Generating themes using inductive coding

Themes	Codes	Freq.
Initiation	Get set go, Task tracking, Setting up	3
Database	Domain model, DB Design, DB edit on the fly	6
Page dev	(PD), Template based PD, Page Linking, Page Header	5
Tables	Data-grid, Intuitive Data-grid, SQL Query, Data-grid add on	4
Self-reflection	Visualizing, Multiple modules, Self suggestions, Refer past	4
User roles	User Role, Role based Access	3
Error handling	security, missing items	2
Business logic	search group, Jumping b/w pages, Tough feature dev	5
Data entry	Master Data entry	3
Go live	testing, Code commit, Deploy, Sharing	9
Usability	Problem, Accidental clicks, Enhanced U, Easy manipulation	6
Styling	CSS, Pretty pages, CSS- good old times, New fancy thing	7
Feedback	Feedback, Work on feedback	3

3.3.1 Data Collection

This study used a pre-validated survey instrument to collect the data. The survey was based on an existing instrument, "USE Questionnaire: Usefulness, Satisfaction, and Ease", that was developed by Lund (2001). Although other survey instruments such as Davis (1989) and Lewis (1995) are widely used across the industry, they concentrated more towards evaluating a interface rather than a tool and hence considered not as good a fit for the purpose of this study.

Arnold Lund developed a tool to deal with questions related to usability while working at Sapient. USE stands for Usefulness, Satisfaction, and Ease of use. These factors are correlated, they are essential in driving user satisfaction and frequency of use (Lund, 2001).

USEFULNESS		1	2	3	4	5	6	7	NA
1. It helps me be more effective. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
2. It helps me be more productive. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
3. It is useful. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
4. It gives me more control over the activities in my life. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
5. It makes the things I want to accomplish easier to get done. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
6. It saves me time when I use it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
7. It meets my needs. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
8. It does everything I would expect it to do. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
EASE OF USE		1	2	3	4	5	6	7	NA
9. It is easy to use. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
10. It is simple to use. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
11. It is user friendly. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
12. It requires the fewest steps possible to accomplish what I want to do with it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
13. It is flexible. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
14. Using it is effortless. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
15. I can use it without written instructions. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
16. I don't notice any inconsistencies as I use it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
17. Both occasional and regular users would like it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
18. I can recover from mistakes quickly and easily. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
19. I can use it successfully every time. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
EASE OF LEARNING		1	2	3	4	5	6	7	NA
20. I learned to use it quickly. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
21. I easily remember how to use it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
22. It is easy to learn to use it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
23. I quickly became skillful with it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
SATISFACTION		1	2	3	4	5	6	7	NA
24. I am satisfied with it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
25. I would recommend it to a friend. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
26. It is fun to use. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
27. It works the way I want it to work. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
28. It is wonderful. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
29. I feel I need to have it. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>
30. It is pleasant to use. <input type="checkbox"/>	strongly disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	strongly agree <input type="radio"/>

List the most **negative** aspect(s):

-
-
-

List the most **positive** aspect(s):

-
-
-

Figure 3.2. USE Questionnaire

The survey consists of 30 close-ended questions, along with two open-ended questions. The close-ended questions focus on perceived usefulness and perceived ease of use along with perceived ease of learning and perceived satisfaction. These questions are scored using a seven-point Likert-type scale from 1- Strongly disagree to 7- Strongly Agree. Additionally, the open-ended questions provide additional qualitative data on experts' opinions about the positive and negative aspects of the tool.

The survey was created using Qualtrics as it is one of the popular online survey tools that is freely available to students of Purdue University.

3.3.2 Recruitment Strategy and Protocol

The study required feedback from experts that have experience with Mendix. While surveying the possible platforms available for getting feedback, the researcher considered LinkedIn (4000+ members), Meetup groups (100+ members), Reddit (80+ members) and the Mendix community forum (60,000+ members). Due to potential limitations in contacting people on LinkedIn and very low outreach in meetup groups and Reddit, the study focused on getting the data from the Mendix community forum, which had a huge active contributor population. The survey was posted along with a brief description of the research and specified that their feedback would remain anonymous.

3.3.3 IRB Protocol

As the study involved human subjects, IRB Protocol was followed for the survey. Participation in the survey was voluntary. Participants had the right to withdraw their participation at any time. No personally identifying information was collected and the survey answers were kept anonymous. The application for IRB exemption was filed on 28th January. IRB approval is included in Appendix B.

3.4 Data Analysis

A mixed approach was followed to analyze the usefulness, ease of use, ease of learning and satisfaction of Mendix experts, and the perceived usability of the system were analyzed using the data collected by the survey. Both qualitative and numerical data was collected through the survey. This data was analyzed using descriptive statistics to gain insights on the perceived usability of the system. The data was also compared and contrasted with the experience of the researcher during the development of the judging application to find commonalities and differences from a usability perspective.

Since the data collected was ordinal and not continuous, hence T - test was not applicable in this case (Allen & Seaman, 2007; Boone & Boone, 2012). Descriptive statistics were used to represent the quantitative data. The open-ended questions were analyzed on the basis of their content and categorized into groups.

3.5 Summary

This chapter described the methodology for the research by outlining the study design and summarizing each individual aspect in the process. The study was conducted at Purdue University during Spring 2019.

CHAPTER 4. IMPLEMENTATION

This chapter describes different aspects of implementation of judging application and the usability survey. Initially the details on the development of TECHFIT judging application have been described and then the further sections describe the implementation of the usability survey.

4.1 Judging app development

The new TECHFIT judging application was developed using Mendix. As mentioned earlier the old application was used as a prototype to help define the requirements of the new application. Various pages of old application were studied in order to chart the relevant modules required to be built in the new application.

4.1.1 Module development

Based on the analysis of the prototype application, the development of the new application was initially sub-divided into the following modules:

1. Admin - Judging app builder
2. Admin - Group management
3. Admin - User management
4. Admin - Judge score management
5. Judge - Score rating

The visual comparisons of user screens of the current ASP.Net (prototype) application and the researcher-developed Mendix application have been provided in Appendix E.

4.1.2 Task tracking

The online interface of Mendix was designed to work as a story boarding dashboard. Agile methodology was followed for developing the application and progress was noted per task. The tasks were classified in the following two broad categories:

- Feature: Unit of functionality that is part of requirement.
- Bug: Failure to meet expectation based on requirement.

The task status was updated as the development progressed. Moreover, the feedback items from advisor were incorporated in the list as bugs or features (Refer figure 4.1).

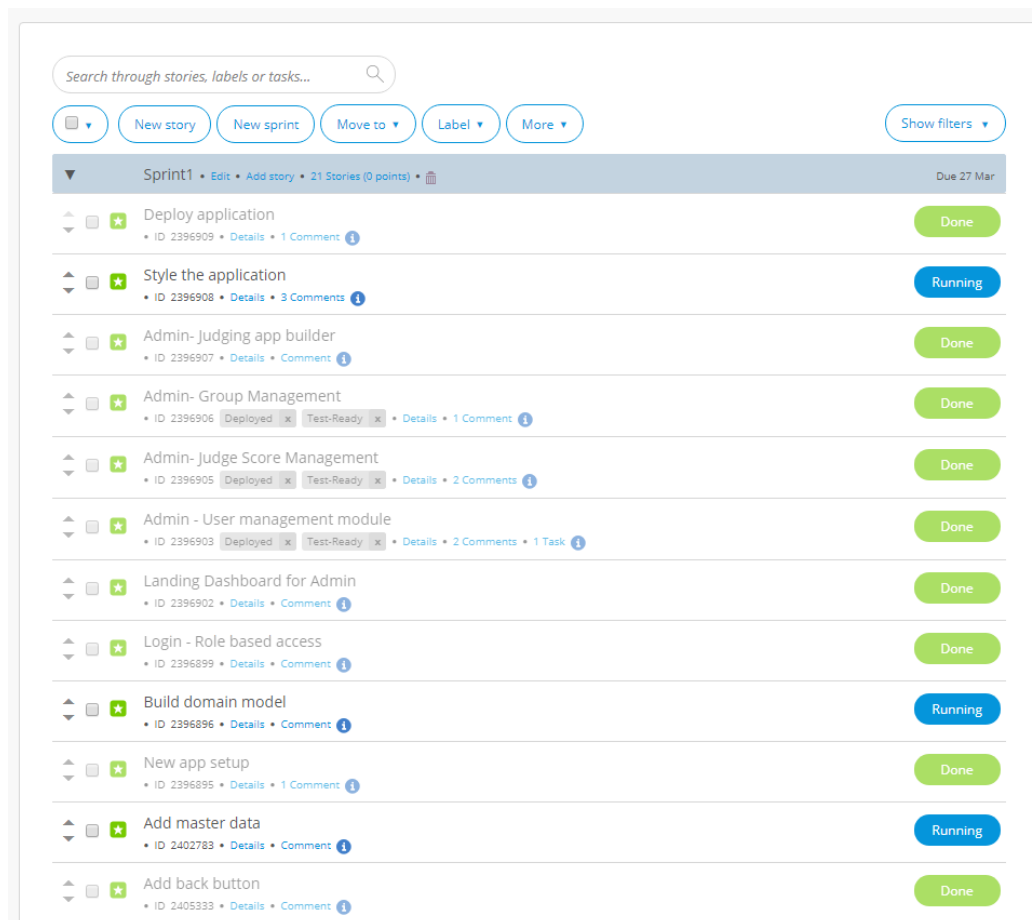


Figure 4.1. Story dashboard with list of tasks and their status

4.1.3 Deployment

The TECHFIT judging application has been deployed on Mendix cloud - free apps. The free deployment is available only for applications up to 100 MB. The application can be accessed on the following link:

<https://techfitjudgingapp-sandbox.mxapps.io/login.html>

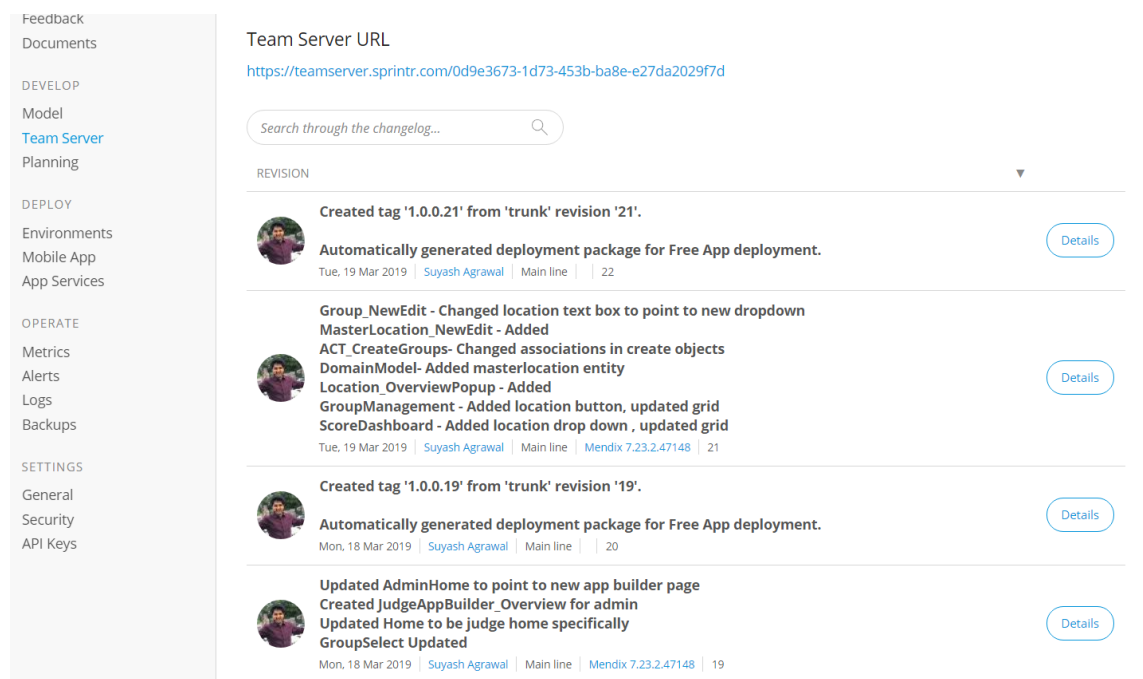


Figure 4.2. Mendix commit history

Figure 4.2 shows a section of the commit history. The first and the third entries show the generation of deployment packages at different times i.e. when the app on the cloud was updated with the current version. The second and fourth entries show the details on code commits during the development process.

4.1.4 Look and Feel

Multiple edits were made in the process of development to have a consistent and simple user interface. While the overall application styling was set to the default Mendix theme "atlas ui", the researcher added additional styling to the elements to suit the application and thus enhance the look and feel of the application.

Moreover, the application was built in a way that it is responsive and thus can be used on mobile devices as well. Figure 4.3 shows the application as seen on a mobile device.

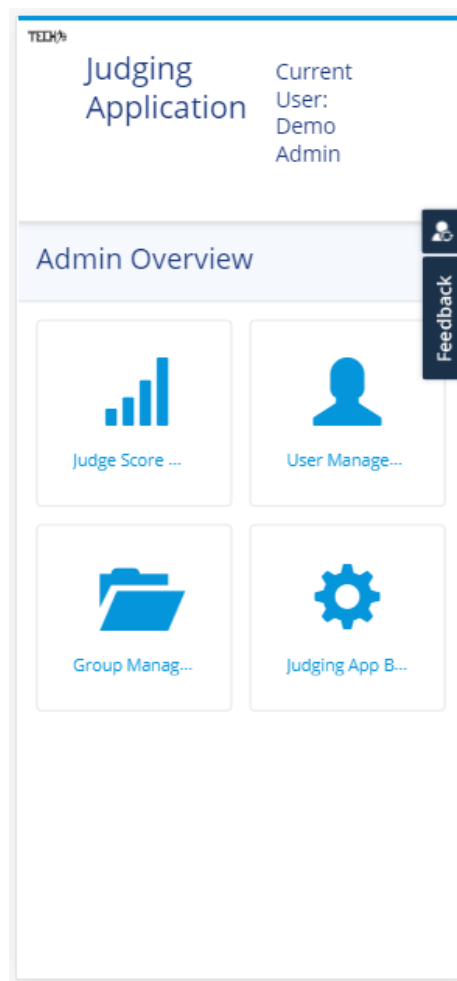


Figure 4.3. Mobile view of new application

4.2 Usability survey

The usability survey sent out to the experts consisted of the following questions.

Table 4.1. Survey Questions

Category	Question	Answer type
Consent	Agreement to participate	Agree/Disagree
Demographic	Gender; Age group	Close ended
	Educational background	Close ended
	Current employment status	Close ended
	Frequency of usage	Close ended
	Experience duration	Close ended
Ease of learning	I learned to use it quickly	Likert scale (1-7)
	I easily remember how to use it	Likert scale (1-7)
	It is easy to learn to use it	Likert scale (1-7)
	I quickly became skillful with it	Likert scale (1-7)
Ease of use	It is easy to use	Likert scale (1-7)
	It is user friendly	Likert scale (1-7)
	It requires the fewest steps possible	Likert scale (1-7)
	It is flexible	Likert scale (1-7)
	Using it is effortless	Likert scale (1-7)
	I can use it without written instructions	Likert scale (1-7)
	I don't notice any inconsistencies	Likert scale (1-7)
	Occasional & regular users would like it	Likert scale (1-7)
	Can recover from mistakes quickly easily	Likert scale (1-7)
	I can use it successfully every time	Likert scale (1-7)
Usefulness	It helps me be more effective	Likert scale (1-7)
	It helps me be more productive	Likert scale (1-7)
	It makes the things to accomplish easier	Likert scale (1-7)
	It saves me time when I use it	Likert scale (1-7)
	It meets my needs	Likert scale (1-7)
Satisfaction	It does everything I would expect it to do.	Likert scale (1-7)
	I am satisfied with it.	Likert scale (1-7)
	I would recommend it to a friend.	Likert scale (1-7)
	It is fun to use	Likert scale (1-7)
	It works the way I want it to work.	Likert scale (1-7)
	I feel I need to have it.	Likert scale (1-7)
	It is pleasant to use.	Likert scale (1-7)
Negative aspects	List upto 3 negative aspects	Open ended
Positive aspects	List upto 3 positive aspects	Open ended

4.2.1 Qualtrics

The survey was built using Qualtrics. All the questions mentioned in Table 4.1 were added. Once the questions and the flow of the survey were verified by the advisor, the survey was posted (refer figure 4.4) on the following link:

https://purdue.ca1.qualtrics.com/jfe/form/SV_6JqHyBXEvKUJ2vz

0% Survey Completion 1

PURDUE
UNIVERSITY.

English

Welcome to the research study!

I am interested in understanding perceptions of usability of Mendix. You will be presented with information relevant to the usability of Mendix and asked to answer some questions about it. Please be assured that your responses will be kept completely confidential.

The study should take you approximately 7-8 minutes to complete. Your participation in this study is voluntary. You have the right to withdraw at any point during the study, for any reason, and without any prejudice. If you would like to contact the researcher in the study to discuss this research, please e-mail Suyash Agrawal at agrawa80@purdue.edu.

By clicking the button below, you acknowledge that your participation in the study is voluntary, you are 18 years of age, and that you are aware that you may choose to terminate your participation in the study at any time and for any reason.

Please note that this survey will be best displayed on a laptop or desktop computer. Some features may be less compatible for use on a mobile device.

☐ I consent, begin the study

☐ I do not consent, I do not wish to participate

Figure 4.4. Usability survey preview

4.2.2 Mendix forum posting

The Mendix community forum was used to request responses to the survey. The researcher posted his request three times over a period of one week in order to get maximum visibility but restraining from spamming the forum. Though the forum has about 60,000 members, the posts were viewed a total of 216 times. The details of the forum posts are as follows:

1. First post - March 16, 2019 - 94 views
2. Re-post - March 19, 2019 - 52 views
3. Final Re-post - March 22, 2019 - 70 views

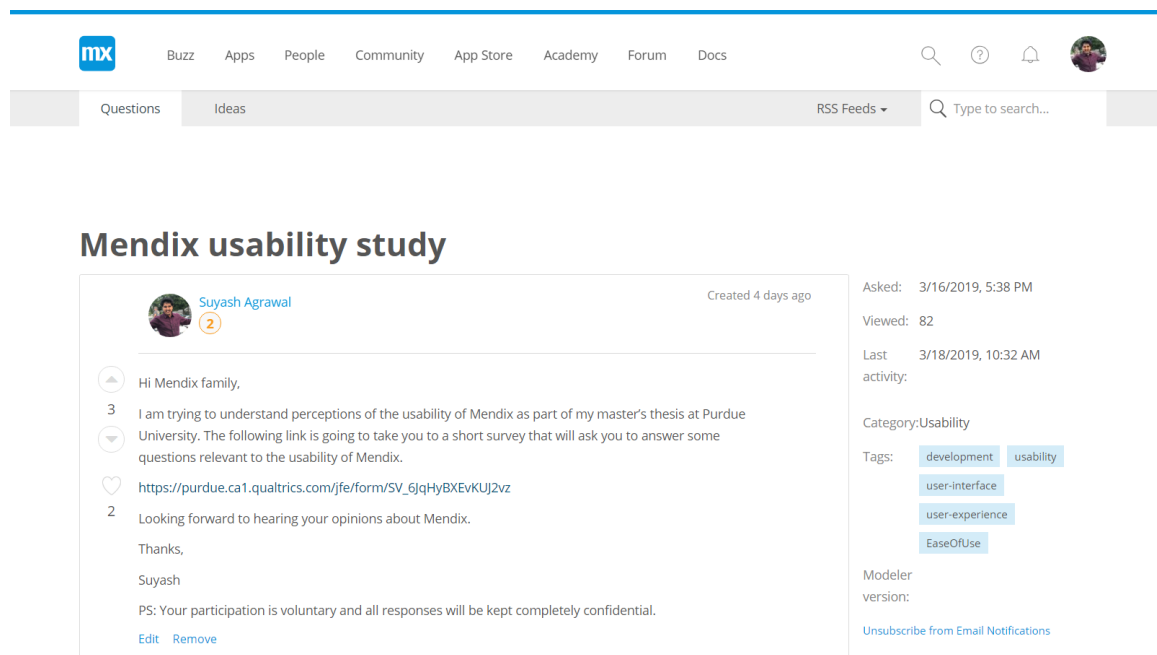


Figure 4.5. Mendix community forum post

4.3 Summary

This chapter had details of how the study was implemented. It included the specifics related to the judging app development and also elaborated on how the usability survey was structured and distributed. The next chapter lists the results of the study.

CHAPTER 5. RESULTS

This chapter presents the results of the overall evaluation performed in the study according to the qualitative and quantitative approaches described in chapter 3. The chapter concludes with a description of relevant findings from the study.

5.1 Thematic analysis of autoethnographic data

As mentioned in section 3.2.3, the data collected from the autoethnography was analyzed using thematic analysis technique. The researcher went through multiple iterations to build the codes that were suitable for the analysis. The following sections provide the analysis for each sub research question.

5.1.1 Analysis using Usability priori codes

In order to answer the first sub-research question, pre-generated codes were used to analyze the data in a better structured way. Inspired from the work of Nielsen (2003) the analysis was based on 5 components i.e. Learnability, Efficiency, Memorability, Errors and Satisfaction, thematic analysis was carried out by coding and categorizing the data on the basis of these 5 themes. Figure 5.1 provides the thematic analysis along with the codes that were generated based on the data.

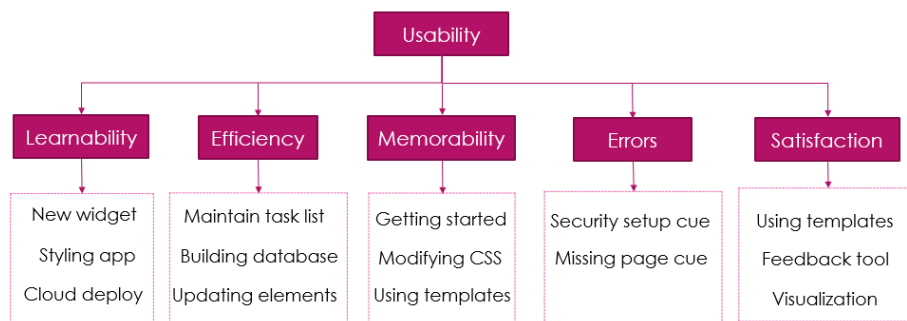


Figure 5.1. Thematic analysis based on usability

The following table 5.1 provides a list of all the references from the data that helped in forming the themes discussed above.

Table 5.1. Thematic analysis based on usability

Code	Data reference
New widget	"...took me five minutes to configure the group box and change the functionality to map to it which was quite cool."
Styling app	"...it is a little tough because Im doing it for the first time but just trying to figure it out by hit and trial is actually fun"
Cloud deploy	"...I'll try to deploy it on and see if it is available online to access. In case this needs to be shared"
Maintain task list	"...in the long run I think tracking what all needs to be done, I think sprinter is going to help a lot."
Building database	"...reworked on the domain model, did few enumerations for session, year etc. basically stuff that I feel would not require entities. I have also changed an association from 1 to many to one to one"
Updating elements	"...it's all just a click away. So that has made a lot of impact in the amount of work that I can get done in a short time"
Getting started	"...Basically, set the context for the application. I've done this before so I know how this will go"
Modifying CSS	"...it is letting me use the same developer tools and the same CSS classes that I used to use in my day-to-day development development in Nokia"
Using templates	"...Then I went ahead & made the admin homepage for which I used the dashboard layout which has four different options"
Security setup cue	"...at this point if you see all the errors that I am seeing this is from what I have changed so far and you see that I did not give permissions for a few pages, microflows and then there are some others like a page is missing a create button or the action button is missing the click"
Missing page cue	
Using templates	"...I went ahead and made the admin homepage for which I used the dashboard layout coz. it will have 4 different options"
Feedback tool	"...later when I ask for the feedback, the feedback tool on this interface would help me a lot"
Visualization	"...a few relationships need to change. Adding a relationship b/w the question & the group would help connect things better "

5.1.2 Analysis based on emotions

One of the sub-research questions aimed at probing the emotional experience of the developer during the app development process. The autoethnographic data was analyzed for direct and indirect emotions represented in each memo. Using Hicks and Hicks (2004) emotional guidance scale, various levels of emotions were grouped together to form 7 major categories of emotions. The autoethnographic data categorized on the basis of different skills/ tasks that were observed during the thematic analysis is represented by Figure 5.2.

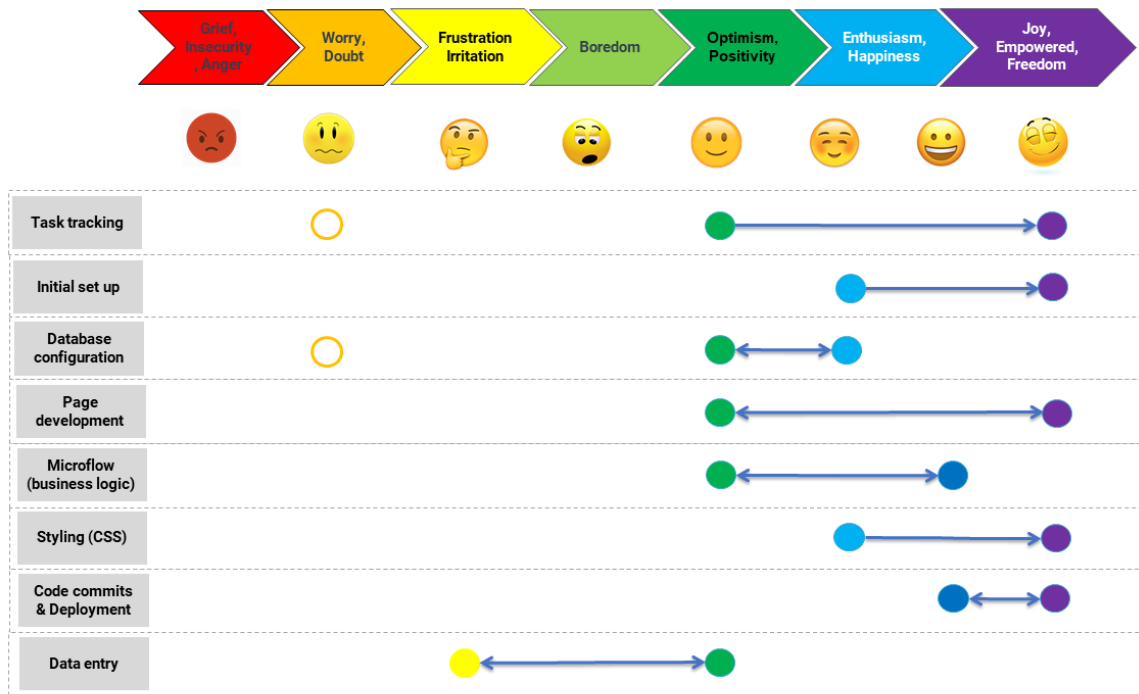


Figure 5.2. Thematic analysis based on emotions

The single sided arrows represent a single directional shift of emotions while double sided arrows represent the emotions varied between the range at different instances during the development process. The solid circles represent the start and end of the emotional journey for a particular kind of task, while the hollow circles represent a single outlier or instance which does not fall into the overall emotional range represented by the data.

It is evident that the overall emotional journey was positive but, there were a few low points in the journey where the developer experienced doubts or worries. The emotions related to task tracking were mostly positive with an exception where the developer was unsure about the value that task tracking is going to add.

”...I have created everything but its kind of useless because I am the only person whos doing it and tracking it, so why would I want to know what has been done from the sprint. (from para 4 in Appendix A)”

The initial setup went smoothly. The database configuration floated between positivity and happiness with an exception where the developer was confused if the database should be built first or built as the development work progresses.

”...I’ve been working on the domain model on paper first and then started building it on the modeler.it has turned out to be a little bit confusing. I thought that I knew all the entities and the attributes that will be required but its been a while now and I am still not done yet. I make these entities, add the attributes and then add associations (which are basically the connections between the entities) but they are a little bit confusing for me right now. (from para 7 in Appendix A)”

The page development and microflow (business logic) related tasks mostly floated between positivity and accomplishment. The styling tasks were particularly between happy and empowered emotional state. The code commits presented the highest level of happiness benchmarks while the data entry was merely a cursory task that needed to be done and was rated in between frustration and optimism.

5.1.3 Analysis for potential roadblocks

The autoethnographic data was run through another round of review, coding and analysis in order to identify any potential points of frustration or roadblocks that were directly present or were latent in the data. Figure 5.3 summarizes the findings.



Figure 5.3. Thematic analysis on points of frustration

As shown in the figure above, two major themes emerged from the thematic analysis of the data. The themes were based on Direct and indirect or latent points of frustration incurred during the app development process.

The direct theme included the frustrations occurring from additional tasks such as master data entry. As evident in para 21 of Appendix A, entering the actual master data in the application was perceived as a boring task. The point of frustration is more evident, when the developer had to re-enter the master data, once the application was deployed on cloud (Refer para 29 in Appendix A). The other sources of frustration were evident in the inability of being able to use a fancy external widget on a page (Refer para 46 in Appendix A) and lack of knowledge of the options available within the tool.

The latent theme included the frustrations that were not directly evident from a particular data point but were hidden behind the data. One of them being the lack of support for Mendix, which was due to the fact that the developer was completely dependent of available documentation and community forum for solving his problems and getting directions and could not turn back to a fellow graduate student for help.

The other aspect of frustration was due to the timeline which required the developer to quickly develop the application while balancing the other day to day activities relevant to a grad student. Juggling between the tasks of a research assistantship, part-time job, developing the application and running the survey made it difficult and frustrating when any aspect of the development got delayed due to any circumstance.

5.2 Survey Data

The results of the survey performed in the study are listed in this section. The first sub-section describes the demographic profile of the participants. The following sub-sections present the outcomes of different aspects of usability such as ease of use, ease of learning, usefulness, satisfaction, positive and negative experiences.

5.2.1 Demographic Profile

A total of 37 respondents completed the survey, including 33 male and 3 female experts as shown in figure 5.4.

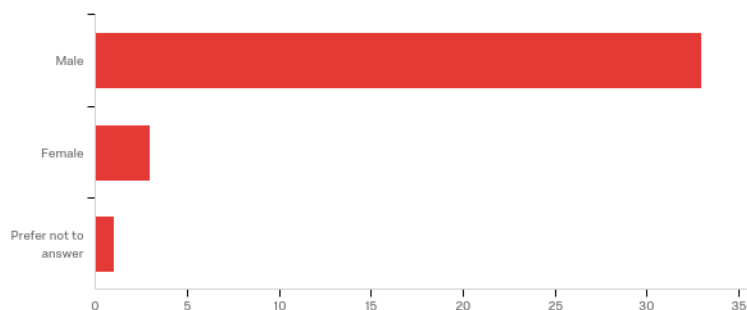


Figure 5.4. Gender distribution

Among the respondents, 89% were male, while 8% were female. Also, most respondents were in between the age group of 18-44. Table 5.2 describes the age group distribution of the participants.

Table 5.2. Age group distribution

Age Groups	Frequency	Percentage
Less than 18	0	0.00%
18-24	8	21.62%
25-34	15	40.54%
35-44	10	27.03%
45-54	2	5.41%
54 and above	2	5.42%

In terms of educational background, 59% of the respondents had a computing background, while 41% of the respondents came from a non-computing background. Figure 5.5 and table 5.3 describes the educational background distribution of the participants.

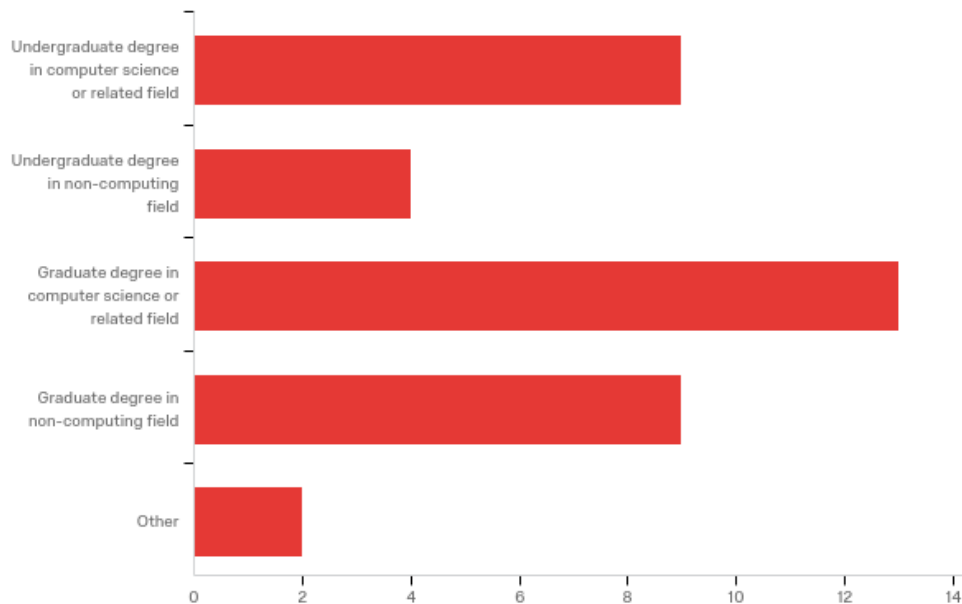


Figure 5.5. Educational background

Table 5.3. Educational background distribution

Highest level education	Frequency	Percentage
Undergraduate degree in computer science or related field	9	24.32%
Undergraduate degree in non-computing field	4	10.81%
Graduate degree in computer science or related field	13	35.14%
Graduate degree in non-computing field	9	24.32%
Other	2	5.41%

It was observed that the two respondents that selected "Other" as their choice come from the following background:

- language agnostic, self taught, multi-skilled
- Secondary education

In terms of current employment status, 78% of the respondents were working as a full-time paid employees as shown in figure 5.6.

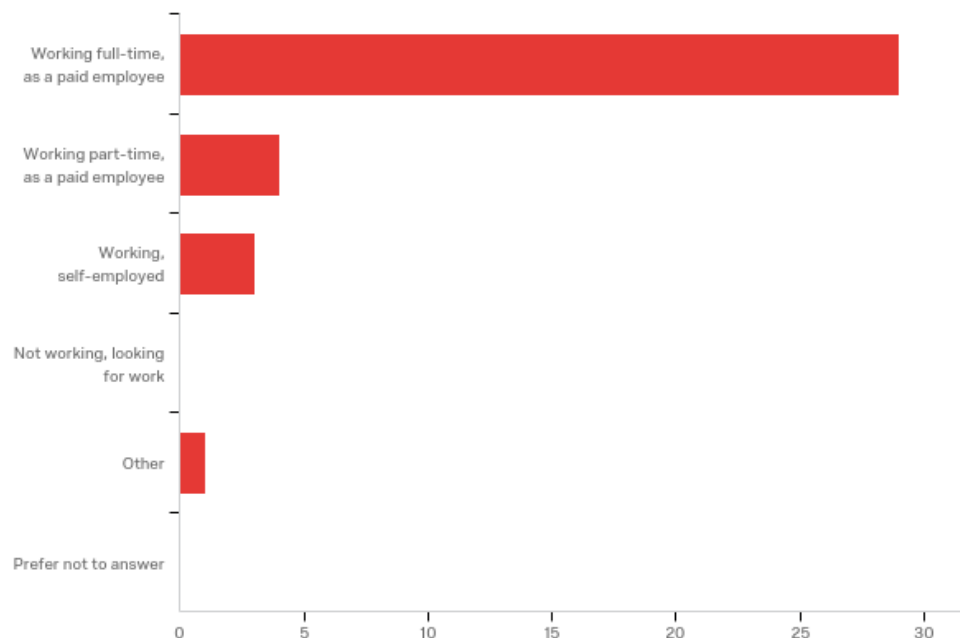


Figure 5.6. Current employment status

89% respondents responded that they use Mendix multiple times per day as shown in figure 5.7. The respondents who said that they rarely used Mendix were excluded from the final analysis as they were redirected to the end of the survey when they selected that choice.

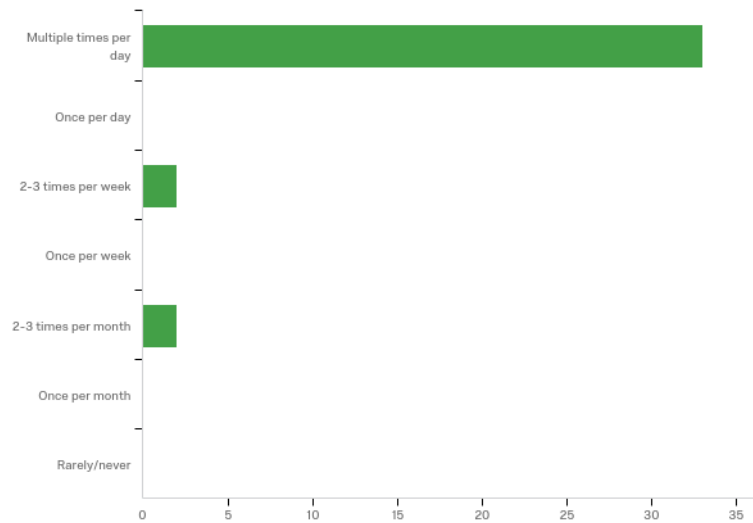


Figure 5.7. Frequency of usage

When asked about how long they have been using Mendix, 62% of the people selected over 24 months while 31% of the people selected less than 1 year experience on it as shown in figure 5.8.

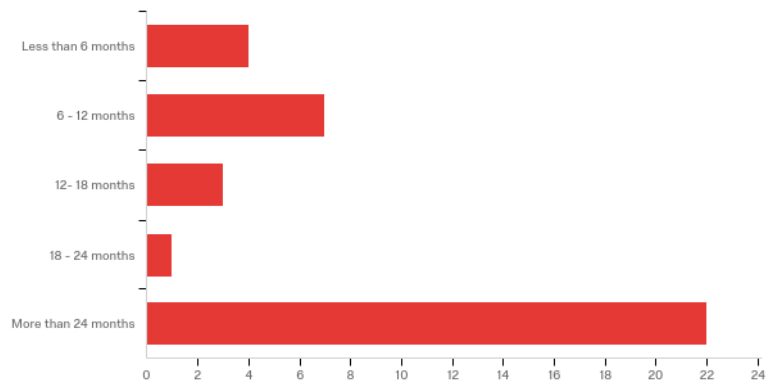


Figure 5.8. Duration of usage

5.2.2 Ease of Learning

Regarding ease of learning, a high percentage of people agreed that Mendix was easy to learn. Table 5.4 shows the various statistics of the data and figure 5.9 represents the show the data in the form of stacked bar chart.

Table 5.4. Ease of Learning

Question	Min	Max	Mode	Median	Disagree	Agree
I learned to use it quickly.	2	7	7	6	10.81%	83.78%
I easily remember how to use it.	3	7	7	6	2.70%	89.19%
It is easy to learn to use it.	1	7	5	5	5.41%	72.97%
I quickly became skillful with it.	1	7	7	6	10.81%	78.38%

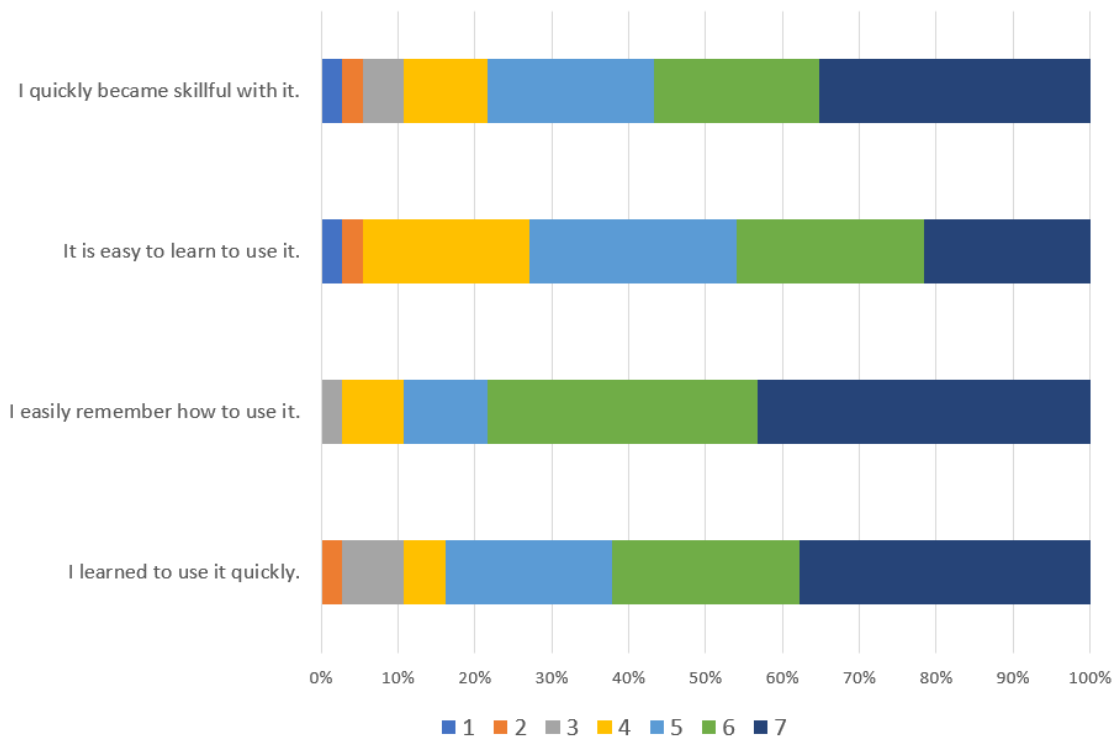


Figure 5.9. Ease of Learning

5.2.3 Ease of use

In terms of ease of use, a high percentage of people agreed that Mendix was easy to use. Table 5.5 shows the various statistics of the data and figure 5.10 represents the show the data in the form of stacked bar chart.

Table 5.5. Ease of use

Question	Min	Max	Mode	Median	Disagree	Agree
It is easy to use	3	7	6	6	5.41%	89.19%
It is user friendly.	1	7	6	6	8.11%	83.78%
It requires the fewest steps possible	1	7	5	5	10.81%	78.38%
It is flexible.	2	7	6	6	8.11%	91.89%
Using it is effortless.	1	7	5	5	29.73%	59.46%
Use it without written instructions	1	7	5	5	32.43%	64.86%
Don't notice any inconsistencies.	1	7	4	4	32.43%	45.95%
Occasional & regular users like it.	1	7	6	6	21.62%	75.68%
Can recover from mistakes easily.	1	7	6	6	8.11%	89.19%
I can use it successfully every time.	1	7	6	6	10.81%	83.78%

5.2.4 Usefulness

In terms of usefulness, a high percentage of people agreed that Mendix was useful. Table 5.6 shows the statistics of the data and figure 5.11 represents the stacked bar chart.

Table 5.6. Usefulness

Question	Min	Max	Mode	Median	Disagree	Agree
It helps me be more effective.	1	7	7	6.5	8.57%	91.43%
It helps me be more productive.	1	7	7	7	8.57%	91.43%
It makes things accomplish easier.	1	7	7	6	8.57%	91.43%
It saves me time when I use it.	1	7	7	6	8.57%	91.43%
It meets my needs.	1	7	6	6	5.71%	91.43%
It does everything I would expect.	1	7	6	6	14.29%	77.14%

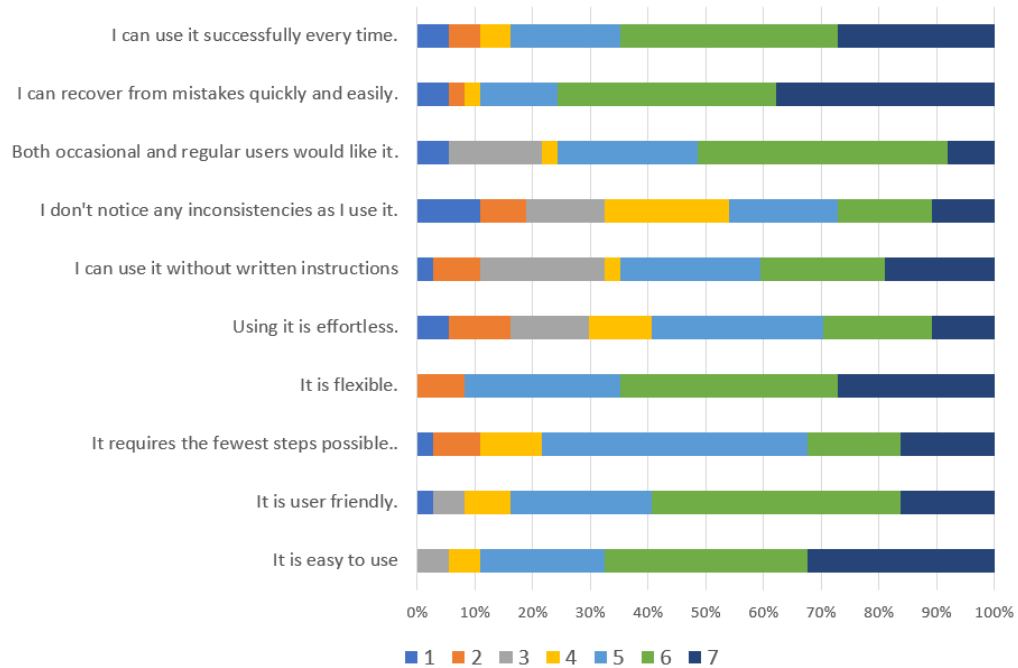


Figure 5.10. Ease of use

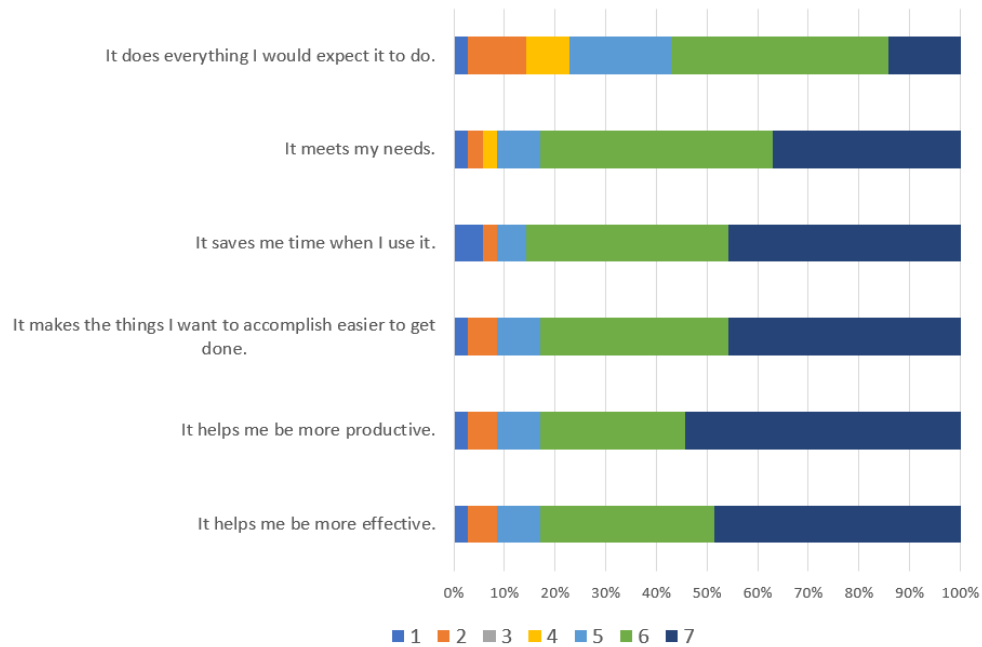


Figure 5.11. Usefulness

5.2.5 Satisfaction

In terms of satisfaction, a high percentage of people agreed that they were satisfied with Mendix. Table 5.7 shows the various statistics of the data and figure 5.11 represents the show the data in the form of stacked bar chart.

Table 5.7. Satisfaction

Question	Min	Max	Mode	Median	Disagree	Agree
Am satisfied with it.	1	7	7	6	8.82%	88.24%
Would recommend it to a friend.	1	7	7	7	8.82%	91.18%
It is fun to use	1	7	7	6	11.76%	82.35%
It works the way I want it to work.	1	7	6	6	14.71%	82.35%
I feel I need to have it.	1	7	6	6	17.65%	64.71%
It is pleasant to use.	1	7	6	6	8.82%	85.29%

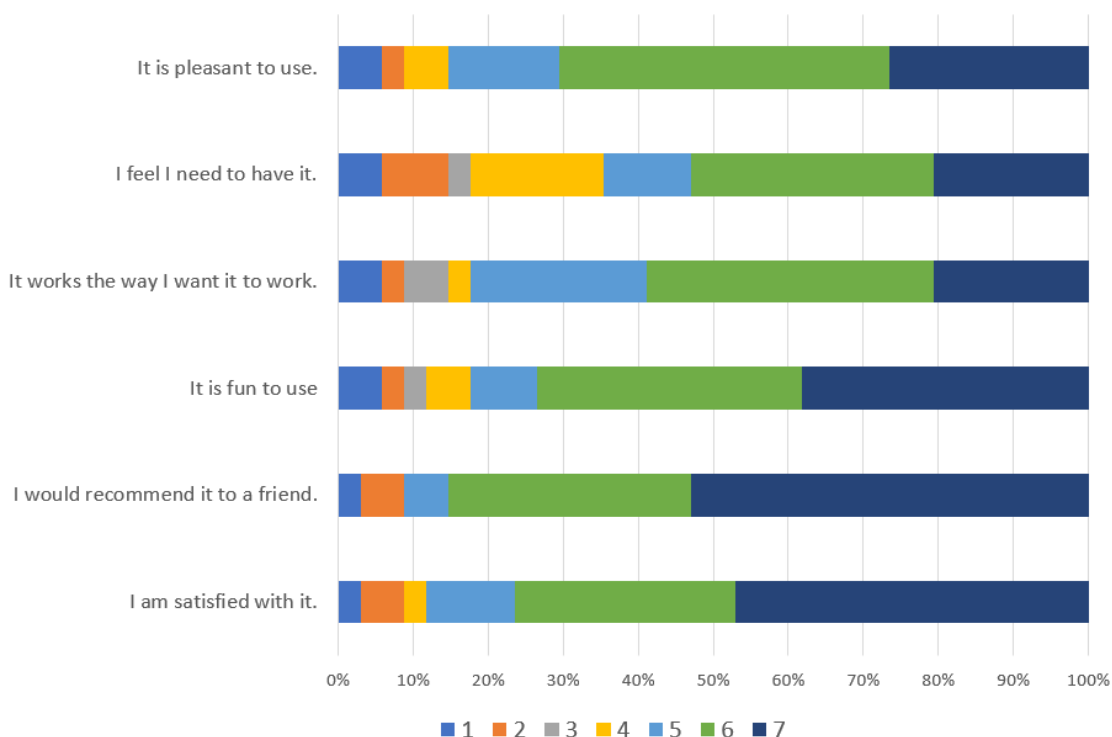


Figure 5.12. Satisfaction

5.2.6 Comparative survey statistics

While looking at the data, the researcher found that the most of the demographics were unusable for the purpose of comparison. The only comparative analysis possible were based on the expert's educational background and experience.

For the purpose of formulation of the table 5.8, the experts with a graduate or undergraduate degree in computing were grouped as Computing and the experts who chose undergraduate or graduate degree in non-computing or other were grouped as Non-computing. It was observed that the computing mean was higher than the non-computing mean for all the cases.

Table 5.8. Comparison of survey data based on educational background

Usability Criteria	Overall mean	Computing mean	Non-computing mean
Ease of learning	5.64	5.96	5.19
Ease of use	5.18	5.40	4.86
Usefulness	5.88	6.15	5.46
Satisfaction	5.59	5.74	5.40

For the purpose of formulation of the table 5.9, all experts with less than 24 months of experience were clubbed together in a group. It was observed that the experience effected their perceptions on ease of learning and satisfaction, but both groups felt the same in terms of usefulness of Mendix.

Table 5.9. Comparison of survey data based on experience

Usability Criteria	Overall mean	Experience <24 months	Experience >24 months
Ease of learning	5.64	5.44	5.78
Ease of use	5.18	5.14	5.21
Usefulness	5.88	5.88	5.87
Satisfaction	5.59	5.37	5.74

5.2.7 Positive Aspects of Mendix

The open ended question on positive aspects of Mendix got 56 responses. The word cloud based on the frequency of the words is depicted below in figure 5.13. The following direct quotes match the theme of this research directly:

”Very easy to create fairly robust applications”

”A lot of basic needs have been properly implemented which lets you focus on your process/application instead of the things like authentication.”

”Smaller gap between developer & product owner (visual representation of code)”



Figure 5.13. Word cloud for positive responses

5.2.8 Negative Aspects of Mendix

The open ended question on negative aspects of Mendix got 56 responses. The word cloud based on the frequency of the words is depicted below in figure 5.14. The following direct quotes match the theme of this research directly:

”Overwhelming amount of boxes on all screens”

”sometimes run into un-explained behavior - mendix generates divs that has a class that changes on click automatically. Then you may have to work around this”

”If apps require tons of custom code, then mendix may not be the best use for these scenarios”



Figure 5.14. Word cloud for negative responses

5.3 Summary

This chapter listed the detailed results of the study. The various sections in this chapter represented the results and findings based on the data collected from autoethnography and the survey. A summarized discussion of findings based on the researcher’s personal experience mixed with the findings from the surveys is presented in the next chapter.

CHAPTER 6. DISCUSSION AND CONCLUSION

This chapter discusses the major findings of the study. Suggestions, potential challenges on using Mendix are provided based on the results from the autoethnographic study and the usability survey. Further section provides the researcher's personal experience with autoethnography. The scope of possible future research is presented towards the end of the chapter.

6.1 Discussion of the Application development experience

As mentioned in Chapter 3 and Chapter 4, a new judging application was developed using Mendix. The application development experience from a developer point of view made the researcher realize a few things which are worth mentioning:

1. Visual coding enhanced the development experience (Refer para 17 of Appendix A and item 48, 55 in part B of Appendix C).
2. Microflows provided the ability of building business logic in the form of flow charts which support thinking in terms of the application flow (Refer figure 6.1 and para 18 of Appendix A).

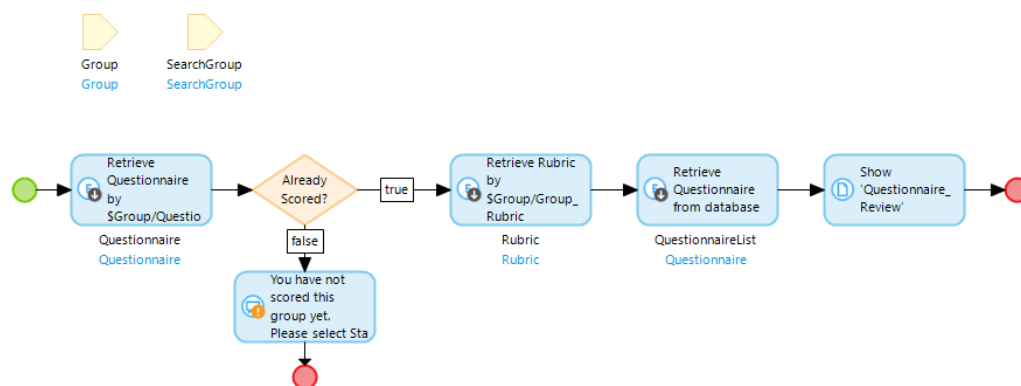


Figure 6.1. Microflow as viewed in Mendix

3. Complex logic hidden under the platform, allowed me to concentrate on building the application rather than getting stuck in complex syntax and configurations (Refer item 42 in part A of Appendix C).
4. Having a positive attitude towards the application along with dedication to build it within the set time line helped maintain personal motivation towards making progress (Refer figure 5.2 thematic analysis of emotions).
5. Code commits and deploying the application provide unsaid sense of satisfaction and accomplishment (Refer figure 5.2 thematic analysis of emotions).
6. Domain model forms a layer of abstraction which enabled me to build the database more like an ER diagram rather than a set of tables (Refer figure 6.1 and para 7, 8, 10, 11 of Appendix A).

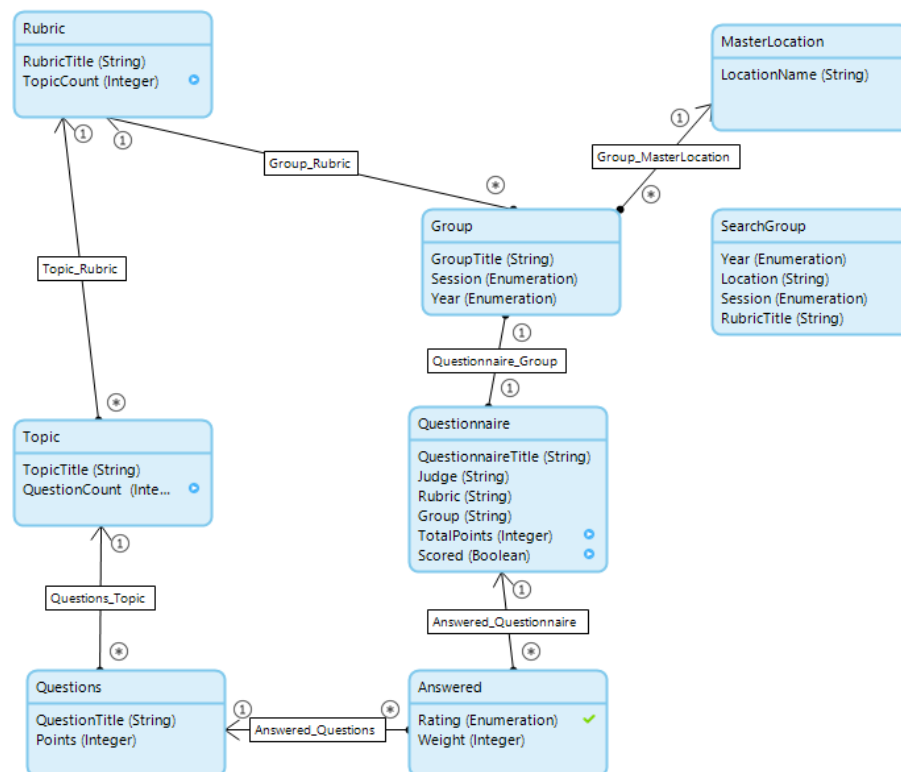


Figure 6.2. Domain model as viewed in Mendix

In order to get a better holistic view, it is important to contrast the app development experience using Mendix to the previous experience of application development using coding. The researcher observed that the visual coding experience is what bolstered the development process. A break from using traditional searching and sorting methods to build the business logic made it look easier to build, understand and follow for the developer. Moreover, a common theme of feeling satisfied and accomplished after code commits can be mapped with the experience of developers in any realm of software development. Thus, RAD has a lot of similarities to traditional software coding techniques but, it is the visual coding aspect that sets it apart.

6.1.1 Potential challenges on using Mendix

The following potential challenges or limitations are evident on the basis of this study:

1. Mendix is not platform independent: It will work only if you have access to a Windows operating system. That adds an additional overhead for developers who have MacBooks (Refer item 46 of Part A in Appendix C) .
 ”Forced to use windows OS”
2. Licenses are costly: Developers can host their applications on free cloud, but that currently has a limitation on number of simultaneous users and memory. Thus, scaling the application commercially would be cost intensive (Refer items 18, 44, 48, 51, 55 of Part A in Appendix C).
 ”expensive for single application...”
 ” its a bit pricey in their cloud license, so its not for smaller companies viable”
3. Keeping up with new versions: Mendix keeps adding new versions every few weeks. Developers need to keep a watch on release notes to check for deprecated methods and fixes from previous versions before updating to a newer version (Refer items 7, 10, 15, 16, 21 of Part A in Appendix C).
 ”Random Bugs that need to get fixed from new releases of the product”

”Frequency of updates makes long term support problematic”

”Quite a lot of fixes in every new Mendix-release”

4. Adding new widgets: There are tons of applications listed on the Mendix app store. Developers keep building new functionalities that can be integrated within any app. The new widget should be checked for documentation, version support and feedback in order to make sure that it would work within your application (Refer item 56 of Part A in Appendix C).

”Can be difficult to decide which widgets to use because of lack of info and support”

It is evident that cost is one of the major challenges or limitations of using Mendix. Multiple experts mentioned that the cost of licenses was very expensive. The other limitation observed in terms of platform-independence is a huge roadblock as it inhibits potential mac users as they would have to go through an additional overhead of configuring a parallel windows version in order to utilize the complete capabilities of Mendix. Another interesting challenge which was faced by the developer himself during the process was about constant new versions. This creates a huge challenge for potential customers or users to keep track of the versions that they use for development. In my personal experience, having to go through a round of sanity testing and reading through the release notes every time a new version is released would add a big red flag to the overall app development experience.

6.1.2 Suggestions/ Recommendations on developing applications using Mendix

The following recommendations for developers were derived from the developer’s experiences of this study:

1. Refer to Mendix documentation: There is ample documentation available on the website to help all levels of developers in configuring various aspects of the application.
2. Get accustomed to the tool: Knowing the basic features and navigation flow of the tool goes a long way in enhancing the development experience.

3. Look for feature options: The visibility, on-click behaviour etc. can be edited in a single click. Follow your intuition of clicking to see the available options.
4. Take a break: Sometimes the mind gets so tired that it stops performing efficiently. Taking a break and starting over again enhances the chances of getting things fixed.
5. Use actual data: Filling actual data helps the developer visualize things better. Actual data helps the developer understand the interactions better.
6. Feedback is important: The tool cannot be gauged without the feedback from the actual user or product owner. The developer might have a skewed understanding of the flow, so getting feedback helps clarify the requirements and enhances the chances of success of the tool.
7. Detailed code commits: The code commits are of prime importance as they not only are a form documentation of what has been done, but also allow the developers to easily reference older iterations.

It is imperative to state that some of the recommendations mentioned above are applicable for all types of application development rather than specific to RAD itself. The suggestions of documenting the code commit, taking a break during long development sessions and getting feedback is more generalized and can be used by any developer irrespective of the development method being followed. However, using mendix documentation as a reference point is strongly recommended as the developer might not have the luxury of discussing the problems with a peer developer. Other exclusive recommendations for Mendix (other RAD tools) developers is to look for feature options and get accustomed to the tool as there is a high level of pre-built customization options available for each entity. Exploring the available customization options before moving on to build business logic helped the developer save crucial time in the overall process.

6.2 Discussion of the survey data

As mentioned in Chapter 5 section 2, the survey data consisted of various demographic aspects of the experts. It tried to capture their perceptions on the usability of mendix by using 26 likert scale based questions on the themes of ease of use, ease of learning, satisfaction and usefulness. The survey also allowed experts to express the most negative and positive aspects of using Mendix. Though the relatively low number of responses makes the possibility of correct generalization low, but the following points are significant:

1. The overall rating on usability of Mendix is over 5 which means that the experts agree on usability of Mendix (Refer Table 5.8).
2. The experts provided 56 responses to negative aspects of using Mendix but still rated it highly. This shows that there is a potential room for improvement in Mendix platform. It is also evident that these aspects are majorly business related and only a few of them are related to the usability of Mendix (Refer Table 5.8, Figure 5.14 and Appendix C).
3. Nearly 40% of the experts come from a non-computing background. It is interesting to see that a large proportion of people who did not have any computing background were able to use Mendix and considered it useful. Furthermore, the analysis of the data shows that the mean scores of non-computing background experts are not very different from the computing background experts. (Refer Table 5.3 and Table 5.8)
4. Nearly 40% of the experts had less than 24 months of experience with Mendix. It is interesting to see how Mendix is relatively new in the domain of application development. Furthermore, the analyses of the data shows that the mean scores for experts with over 24 months experience is not very different from the experts with less than 24 months of experience. Thus, Mendix was considered fairly useful by experts with all levels of experience. (Refer Table 5.9)

6.3 The power of the journey - self realization

This study used autoethnography as a first-person research method for collecting and analyzing the qualitative data on developing an application on a RAD tool. Though the purpose of autoethnography was to be able to evaluate the researcher's lived experiences and get the researcher's perspective on Mendix, the process of undergoing the study (Refer Chapter 4) and looking back at the collected data (Refer Appendix A) made the researcher realize that the visual aspects of the RAD tool greatly enhanced his experience of development. The researcher's liking towards styling the application was evident as the researcher went an additional mile to make the application look better.

Software developers across the industry are often distinguished as front-end and back-end developers. There is another distinction which separates the coders from the non-coders. The study made the researcher realize that he would not be suited for non-visual coding job and hence looking for a normal software developer job might not be fruitful in the long run.

Another realization during the process was about the feeling of getting conscious of yourself and your surroundings when you are being filmed. Taking inspiration from Heshmat, Neustaedter, and DeBrincat (2017)'s work where an always-on video recording system for home was used to capture the day to day moments, the researcher tried to capture self pictures using the webcam of the laptop. The intention was to capture the facial expressions during the development process which could be compared with development screenshots to gain insights on information that was not captured in the voice notes. It was observed that it made the researcher so conscious about his looks and cleanliness of the room, that it was inversely affecting the development experience.

6.4 Future scope

There are a few ways in which future work could build on the work done in this study. As a limited amount of time was available to conduct the study, few aspects of the application development were deferred for future works. Populating available options dynamically for judges and adding visual representation for scores in the form of charts are two suggested future works for the judging application.

As per the literature, most academic organizations and institutes recruit students who can help develop or maintain software to support their activities. Various institutes such as Stanford, Columbia University and Cornell have various research assistant positions open for undergraduate and graduate students that can work in projects requiring various levels of software development skills (*Fall Research Assistant Announcement*, n.d.; Stanfordvptl, 2017; Weldon & Reyna, 2015). Though in the cases mentioned above, students need to be well-versed in the technology to be selected for the positions, certain universities run programs such as Undergrad Research Assistant Program (URAP) where inexperienced students are given research assistantships to work on real projects with faculty mentors (*Office of Undergraduate Research*, n.d.). Thus students need to develop software development skills from the beginning in order to be successful. Moreover, the universities should also be open to include new popular technologies such as Mendix (RAD in general) in their curriculum. This would give students a chance to experience different aspects of IT and then make an informed decision on what they would like to pursue.

The above study has tried to provide a case for faculty to add RAD as part of their curriculum. This can possibly be done in two ways. First, incorporating a new RAD course which would allow students from computing and non-computing background explore the possibilities in the sphere of rapid application development. The second way is to include Mendix as a RAD tool to supplement the learning experience in a software development life cycle or project management course. This would enable the new developers to apply and experience the various aspects of the software development life cycle in a real environment. It would also help people with experience on using traditional

coding based development to be exposed to a new technology available in the market which can ultimately help them in taking an informed decision about what they would like to pursue. As the data shows that Mendix is easy to learn and easy to use, both professors and students would be able to use it easily.

The findings from the study provide a case to the people working at Mendix to look into possible business decisions that could potentially benefit them. The positive feedback from the experts provides a positive reinforcement for them to continue what they have been doing. However, the negative feedback provides important points on the potential problems and roadblocks that Mendix developers have to face which effect their feelings about the tool. It is needless to say that the feedback could help the Mendix team to improve their tool and give them a chance to have greater success in the RAD domain.

In conjunction with the above suggestion, the researcher also recommends using the survey from this study to get feedback from students to compare their experience with the experts. Getting feedback on the usability of Mendix from students who are relatively new to the tool would provide a good point of comparison between the perceptions of experts and beginners.

As literature shows that autoethnography is a relatively lesser known technique of qualitative analysis. The potential limitations and suggestions from this study can be used for future software development related to autoethnographic studies. The methods followed to collect and analyze the data could be modified and improved as per requirement. This would enable future researchers to take a better-informed decision about using this approach in their research study.

REFERENCES

- Affairs, A. S. P. (2013). *Judging the severity of usability issues on web sites: This doesn't work*. Department of Health and Human Services. Retrieved from <https://www.usability.gov/get-involved/blog/2005/10/judging-the-severity-of-usability-issues.html>
- Albert, W., & Dixon, E. (2003). Is this what you expected? The use of expectation measures in usability testing. In *Proceedings of the Usability Professionals Association 2003 Conference, Scottsdale, AZ*.
- Allen, I. E., & Seaman, C. A. (2007). Likert scales and data analyses. *Quality progress*, 40(7), 64–65.
- Beck, K., & Gamma, E. (2000). *Extreme programming explained: embrace change*. Addison-Wesley Professional.
- Begel, A., & Nagappan, N. (2007). Usage and perceptions of agile software development in an industrial context: An exploratory study. In *Empirical software engineering and measurement, 2007* (pp. 255–264).
- Berger, H., & Beynon-Davies, P. (2009). The utility of rapid application development in large-scale, complex projects. *Information Systems Journal*, 19(6), 549–570.
- Bevan, N. (1991). What is usability. In *Proceeding of the Fourth International Conference on HCI*.
- Beynon-Davies, P., Mackay, H., & Tudhope, D. (2000). Its lots of bits of paper and ticks and post-it notes and things...: a case study of a rapid application development project. *Information Systems Journal*, 10(3), 195–216.

- Boone, H. N., & Boone, D. A. (2012). Analyzing likert data. *Journal of extension*, 50(2), 1–5.
- Bot, R. L. (2013). *A study of perceptions, usability and future adoption of an electoral database*. Unpublished doctoral dissertation, Purdue University.
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2), 77–101.
- Brooke, J., et al. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 189(194), 4–7.
- Brooks, F., & Kugler, H. (1987). *No silver bullet*. April.
- Cecchinato, M. E., Cox, A. L., & Bird, J. (2017). Always on (line)?: user experience of smartwatches and their role within multi-device ecologies. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (pp. 3557–3568).
- Chandler, D., & Munday, R. (2011). *A dictionary of media and communication*. OUP Oxford.
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, 319–340.
- Desjardins, A., & Ball, A. (2018). Revealing Tensions in Autobiographical Design in HCI. In *Proceedings of the 2018 on Designing Interactive Systems Conference 2018* (pp. 753–764).
- Dissanayake, N. R., & Dias, G. (2014). Best practices for rapid application development of AJAX based Rich Internet Applications. In *2014 14th International Conference on Advances in ICT for Emerging Regions (ICTer)* (pp. 63–66).
- Ellis, C., Adams, T. E., & Bochner, A. P. (2011). Autoethnography: an overview. *Historical Social Research/Historische Sozialforschung*, 273–290.

Ellis, C., & Bochner, A. P. (2000). Autoethnography, personal narrative, reflexivity: Researcher as subject.

Fall research assistant announcement. (n.d.). Retrieved from
<http://www.earth.columbia.edu/articles/view/2573>

Gottesdiener, E. (1995). RAD realities: Beyond the hype to how RAD really works. *Application Development Trends*, 2(8), 28–38.

Harriger, A. R., Harriger, B., Flynn, M., & Flynn, S. (2015). Attracting Minorities to ET through TECHFIT. In *2015 asee annual conference and exposition*.

Heshmat, Y., Neustaedter, C., & DeBrincat, B. (2017). The autobiographical design and long term usage of an always-on video recording system for the home. In *Proceedings of the 2017 Conference on Designing Interactive Systems* (pp. 675–687).

Hicks, E., & Hicks, J. (2004). *Ask and it is given*. Hay House, Inc.

Höök, K. (2010). Transferring qualities from horseback riding to design. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries* (pp. 226–235).

Hotle, M. (2014). The Disintegration of AD: Putting it Back Together Again. *PDF*.
<http://www.gartner.com.br/EnterpriseIntegrationSummit,SaoPalo,Brazil:GartnerGroup.Retrieved,1>

Howard, A. (2002). Rapid application development: Rough and dirty or value-for-money engineering? *Communications of the ACM*, 45(10), 27–29.

Jones, G., & Leung, V. (2005). Visual surveillance: a systems engineering approach for rapid development.

- Kasunic, M. (2005). *Designing an effective survey* (Tech. Rep.). Carnegie-Melon Univ Pittsburgh PA Software Engineering Institute.
- Kerr, J., & Hunter, R. (1993). How to build a fully functional system in 90 days or less. *How to Build a Fully Functional System in 90 Days or Less*, 87.
- LeCompte, M. D., Schensul, J. J., et al. (2012). *Analysis and interpretation of ethnographic data: A mixed methods approach* (Vol. 5). Rowman Altamira.
- Lewis, J. R. (1995). IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7(1), 57–78.
- LinÅker, J., Sulaman, S. M., de Mello, R. M., Hst, M., & Runeson, P. (2015). Guidelines for conducting surveys in software engineering. *Technical report*.
- Lucero, A. (2018). Living without a mobile phone: an autoethnography. In *Proceedings of the 2018 on Designing Interactive Systems Conference 2018* (pp. 765–776).
- Lucero, A., Desjardins, A., Neustaedter, C., Höök, K., Hassenzahl, M., & Cecchinato, M. (2019). A Sample of One: First-Person Research Methods in HCI.
- Lund, A. M. (2001). Measuring usability with the USE Questionnaire¹². *Usability interface*, 8(2), 3–6.
- Maréchal, G. (2010). *Autoethnography In AJ Mills*. G.
- Martin, J. (1991). *Rapid application development*. Macmillan Publishing Co., Inc.
- Marvin, R. (2018, Aug). *The best low-code development platforms of 2018*. PCMAG.COM. Retrieved from <https://www.pcmag.com/roundup/353252/the-best-low-code-development-platforms>

- Maximilien, E. M., & Williams, L. (2003). Assessing test-driven development at IBM. In *25th International Conference on Software Engineering, 2003. Proceedings* (pp. 564–569).
- Medford, K. (2006). Caught with a fake ID: Ethical questions about slippage in autoethnography. *Qualitative Inquiry*, 12(5), 853–864.
- Nielsen, J. (2003). *Usability 101: Introduction to usability*.
- Office of undergraduate research*. (n.d.). Retrieved from <http://undergradresearch.northwestern.edu/open-positions-undergraduate-research-assistant-program-urap>
- Rapley, K. (1995). RAD or TRAD or both? The future of software development. In *IEE Colloquium on Will Tickit and ISO 9000 Survive Rapid Application Development?* (pp. 3–1).
- Root, R. W., & Draper, S. (1983). Questionnaires as a software evaluation tool. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (pp. 83–87).
- Rosenberg, D., & Stephens, M. (2008). *Extreme programming refactored: the case against XP*. Apress.
- Schneider, D. (2018). *Usability and user experience surveys*. Retrieved from http://edutechwiki.unige.ch/en/Usability_and_user_experience_surveys
- Smith, A. R., Colombi, J. M., & Wirthlin, J. R. (2013). Rapid development: A content analysis comparison of literature and purposive sampling of rapid reaction projects. *Procedia Computer Science*, 16, 475–482.
- Stanfordvptl. (2017, Apr). *Student software developer*. Retrieved from <https://vptl.stanford.edu/job/student-software-developer>

- Suri, P., Kumar, S., & Singh, G. (2011). Precision in rapid application development and reusability of software components for greater performance using ranking mechanism. *International Journal of Computer Applications*, NY, 35(8), 21–27.
- University program - free application development software resources*. (n.d.). Retrieved from <https://www.mendix.com/university-program/>
- Weldon, R. B., & Reyna, V. F. (2015). How to successfully incorporate undergraduate researchers into a complex research program at a large institution. *Journal of Undergraduate Neuroscience Education*, 13(3), A192.
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). *Experimentation in software engineering*. Springer Science & Business Media.
- Xiao, L., & Dasgupta, S. (2005). User satisfaction with web portals: An empirical study. In *Web systems design and online consumer behavior* (pp. 192–204). IGI Global.
- Zazelenchuk, T. W., & Boling, E. (2003). Considering user satisfaction in designing web-based portals. *Educause Quarterly*, 26(1), 35–40.

APPENDIX A. AUTO ETHNOGRAPHIC NOTES

Task in Hand: Making user stories on sprintr

Feelings: Confident, thinking

Recording: Before I start making tasks on sprintr

Inductive codes: Task tracking

Transcript:

This is how I can track all the things that are required for my application. I have not followed it from the beginning because I thought it would not be required as I am building all of the things myself. But a lot of things keep crossing my mind that need to be done or may be good to have and I need a way to basically keep a track of all those tasks. So, I started making stories. I think this will make my life easier. Another reason behind this is that later when I ask Prof. Alka for the feedback, the feedback tool on this interface would help me a lot. So, in the long run I think tracking what all needs to be done before considering this application as complete, I think sprintr is going to help a lot. [1]

I would also note that it's a little difficult to make all the stories yourself. It's generally the scrum master or the product owner that makes all the stories. You might consider me as the product owner in this case but it's just a little hard to figure out how exactly to keep those descriptions non-technical. [2]

Task in Hand: Making user stories on sprintr Contd.

Feelings: Thinking, processing, positive

Recording: Between the creation of different two stories

Inductive codes: none

Transcript:

I just realized that it is difficult to understand how much detail needs to go in each user story so that you know it's understandable by somebody looking at it for the first time. Though, it's all in my mind because of seeing the prototype application a lot of times but figuring out a way to mention everything in common words is a little difficult. So, I am a little bit confused (in a good way) trying to make sense of what all needs to go inside this. [3]

Task in Hand: Making user stories on sprintr Completed.

Feelings: Accomplished, relieved, confused, happy

Recording: After the creation of stories

Inductive codes: none

Transcript:

So, this has been a productive session. I finally figured out what are user stories needed to be in the sprintr. I have created everything but it's kind of useless because I am the only person who's doing it and tracking it, so why would I want to know what has been done from the sprintr. Right now I don't know if it's going to be very useful but nevertheless I have created everything that I could think of for now so let's see how it goes (happy) [4]

Phase 1 starts:

Task in Hand: Install desktop modeler

Feelings: Neutral, confident

Recording: before installing modeler

Inductive codes: setting up

Transcript:

So, the current task in hand is to basically download the Mendix modeler and make sure that I am able to login to the desktop modeler. Basically, set the context for the application. I've done this before so I kind of know how this will go. [5]

Task in Hand: Install the desktop modeler

Feelings: confused, question, waiting, accomplished, happy, excited

Recording: after installing modeler

Inductive codes: get set go

Transcript:

So, I was able to install the desktop modeler pretty quickly. The only trouble that I had was to find out the link to download the modeler. It probably took like five minutes to download it another five minutes to install it. after that I needed to sign in. I logged into the modeler using my credentials. I was able to see the project that I had created the TECHFIT judging app. I can start the development process now (so excited). [6]

Task in Hand: Build the domain model

Feelings: tired, puzzled

Recording: In between building the domain model

Inductive codes: Domain model,DB Design

Transcript:

So, I have been working on this early morning. I've been working on the domain model on paper first and then started building it on the modeler. it has turned out to be a little bit confusing. I thought that I knew all the entities and the attributes that will be required but it's been a while now and I am still not done yet. I make these entities, add the attributes and then add associations (which are basically the connections between the entities) but they are a little bit confusing for me right now. This is probably the first time I am building an application from scratch all by myself, so that might be one of the reasons. [7]

Task in hand: Start building pages

Feelings: Decisive Ready excited

Recording: before starting to build the dashboard

Inductive codes: DB edit on the fly, page dev

Transcript:

So, the new plan is that I'll just keep building the domain model as I build the pages and see how these pages need those attributes and then figure out what goes where, instead of trying to make the whole domain model in the beginning itself. So, I'm going to start with building the basic judging dashboard that is the place where the admin typically lands once he logs in. so let's do it! [8]

Task in hand: developing initial pages

Feelings: Happy, satisfied

Recording: after building first 2 pages for judge view

Inductive codes: Data grid, intuitive-driving

Transcript:

So I made two pages and this is one of the good things that I want to capture from usability stand point that basically the moment do you add a data grid it automatically adds the new and edit and delete options for that data grid and the good part is that it immediately tells you that that you need to add those pages because the page to that new button does not exist so it's kind of intuitive and is driving me towards what I need to do next. [9]

Task in hand: Refine domain models and build pages

Feelings: Positive

Recording: Domain model and page building in process

Inductive codes: DB edit on the fly, visualizing

Transcript:

So, this is the next session and I got some time to think about the domain model again and I feel that session and location should not be different entities they should probably be a part of the group. So, maybe creating some enumerations in this would help. Also, I think a few relationships need to change. Adding the relationship between the question and the group would probably help us connect things better so I'm gonna go ahead and do that and then try to build forward on the application (right now the feeling is positive). [10]

Task in hand: Update domain model, make admin dashboard

Feelings: satisfied, neutral

Recording: After the dev session

Inductive codes: DB edit on the fly, Template PD; Page linking, user role

Transcript:

So, in this session, I have reworked on the domain model and I did few enumerations for session, year etc. basically stuff that I feel would not require entities. I have also changed an association from one to many to one to one because I think for each group there's going to be just one questionnaire that will exist. Then I went ahead and made the admin homepage for which I used the dashboard kind of layout because it will have four different options, so I have created the basic four pages for score management, user management, group management and judging app builder. [11]

Next, I need to build all the pages that link to the admin dashboard, plus the next thing that I need to figure out is who is to basically be able to view what page. I think we're getting closer to the point where I need to have two different roles and set permissions to all those pages so that I can start viewing two different user views. So, I'll probably have to research a little about it and then get back to work on this over all this. (Overall fun, little confusing but okay). [12]

Task in hand: Enable security roles for Admin and Judge

Feelings: Positive, know what to do

Recording: Before configuring the user roles

Inductive codes: user role, Role based Access

Transcript:

The aim for this session is to enable the security roles for the admin and the judge, so, we have two different landing pages for them and then I think I'll have to set it for all individual pages and the look into the microflows that I've made. So, I'll look into that ultimately. As long as I'm able to login as two different users and see two different pages I think the current task will be done and then I can build a plan from there. [13]

Task in hand: Getting two different landing pages based on role

Feelings: Sad, confused

Recording: after trying to configure security roles correctly

Inductive codes: none

Transcript: So I did not have much success at this time in configuring the user roles. I created two different roles and tried to set the security to map it with domain model and the different pages but I'm not able to get two different pages based on the login role, so I have to do some more research on how that whole thing works and then look into this again. [14]

Task in hand: Get the user role working

Feelings: Frustration, uncertainty, ideation

Recording: in between the development session

Inductive codes: multiple modules open; suggestion to self

Transcript:

I am at the loss of direction right now. Looking at the modules right now I see that a lot of things are open. I have started the group management module, have started the judging module, have done some work on the admin module but right now I am not sure how to proceed because doing it the way I thought did not quite work correctly at that point, but can still bring in some more thoughts and give it a try. [15]

So, what I am thinking right now is to just concentrate on the changes and build judge view and what the judge would see, and build upon the flow that the judge would use typically and go from there and see how things are. (The feeling is a little frustration mixed with the uncertainty. I would say that I'll give some time and it will get sorted out.) [16]

Task in hand: Build judge view

Feelings: Thoughtful, positive

Recording: In between the dev session

Inductive codes: db edit on the fly

Transcript:

A note to Self. I was trying to figure out what was the next step to go from the judge rubric to figuring out what all groups need to be there. I think I need a place to store all the year, location, session and rubrics settings so that I can pull them up as the user specifies what he wants and then move to the next page. So, I have created another search group entity. I think this does not need to be associated with any of the others are attributes or entities are such, but this is what is going to help me. Hopefully. so we'll see how that goes. [17]

Task in hand: Getting the groups for judge populated

Feelings: Easy, fun, confident, happy

Recording: in between the development session

Inductive codes: error handling - security, missing items

Transcript:

So, another note to self for while doing this judge flow. I ended up making multiple microflows and multiple changes in order to figure out the to show the exact groups that would be available for that judge, (so the good thing is that) I think this is going to work but we'll see as it's not complete. But why am I doing this voice note is because at this point if you see all the errors that I am seeing this is from what I have changed so far and you see that I did not give permissions for a few pages and the microflows and then there are some other issues like a page is missing a create button or the action button is missing the click, so it's a different that I kind of know what I need to do to get rid of the bugs this way. [18]

The other good thing that I found was inside the data grid where I was able to write an xpath constraint which is quite self-explanatory. It's kind of easy to follow so for the given session that should be equal to the search group session that the user entered the year should be equal to the search group year that they use it then the rubric title should be equal to the rubric group title that the user selected in the previous page on the pop up so this is basically going to let me narrow down the group which is kind of nice it's like building in SQL query. (just fun stuff but it's a lot easier I think.) [19]

Task in hand: Complete search group functionality

Feelings: Tired

Recording: After quitting the session midway

Inductive codes: logic to search group

Transcript:

So, this was probably the first tiring session. I have made a lot of changes to my logic to accommodate this whole search group thing that they would use to be able to select a year and session and then go to the groups that are available for that judge. I also added logic for the values to basically display on the pop-up. It was working fine before, and the flow is broken now. But I think I am just too tired right now, I am not getting much done this way it's late night so I should probably just keep it for tomorrow. [20]

Task in hand: Fill master data

Feelings: Boring, neutral

Recording: Before filling master data

Inductive codes: data entry

Transcript:

What I'm going to do is to create the master data from the prototype applications. so, I'm going to pick the rubrics and I'm going to see what all topics do they have and what all questions and their scores and just fill in some relevant data here so that when I try to run some logic here I can get some data that would be easier to compare and it would feel more real rather than having those random rubric and group names which did not make sense and felt very weird anyway. It's time consuming and a little boring. [21]

Task in hand: Enter master data

Feelings: neutral

Recording: in between data entry

Inductive codes: testing, flow

Transcript:

So, I'm pausing my data entry in between because I discovered something new. basically, the way I was trying to enter questions the admin would have to go back-and-forth on multiple pages and it was driving me crazy, but I figured that instead of having a separate page layout how about we have a pop-up layout that could just simplify the task for the person. It so feels a lot more intuitive that way though if you need to click save it goes back to the previous page you see you think that's the way to go. So, this is probably the first usability problem that I've seen while actively using this application. [22]

Task in hand: Get user information displayed

Feelings: neutral, confused

Recording: Before working on the logged in user

Inductive codes: page header, past experience

Transcript:

I have spent the night thinking how I will integrate the user to each page or if there is a way to integrate it without having to code it on each page but have direct access like the way it was in past in my previous application but the logic was already built in for that one, I just had to copy the logic typically. So not sure how I can do that, so apart from trying to integrate that flow where I can show the user at each page I also need to figure it out at the page titles and the headers for each page that would be required so let's see how it goes. [23]

Task in hand: Display logged in user on all pages

Feelings: Clueless, neutral, searching, working, neutral

Recording: In the process of adding logged in user snippet

Inductive codes: user in header

Transcript:

So, I searched on my previous project on how the user account thing was done it's a little complicated, but I think it makes sense to do a snippet. I have started making good progress but unfortunately, I need to go out for lunch so I'll probably just come back after an hour and again try to complete the flow. It's taking some time I think if I had started in the beginning with the admin thing then it would have been easier so let's see how it goes. [24]

Task in hand: Filling data and getting the application running

Feelings:

Recording: After the session

Inductive codes: master data update, code commit; deploy, sharing

Transcript:

It was an all afternoon long session. a few things have been accomplished. I have figured out a way to show the current user on the page so now we have a full name that remains constant on all the pages. I have removed some unnecessary information and changed the full names for different logins. So, then I figured that I have all changes on my machine only till now. so, I have all the changes committed all the changes so now they were like 35 changes but I think this is the first commit so I have tried to mention the things that I've committed in this change. [25]

I'll try to deploy it on and see if it is available online to access. in case this needs to be shared with Prof. Alka or anybody else then I can just go ahead and do that. This day overall has been a little discomfoting or unsettling if you would say but ultimately as long as I'm able to figure out the logic, it makes me happy at the end of the day. [26]

Task in hand: Filling master data

Feelings: Confident,oops, positive

Recording: While filing the master data

Inductive codes: Testing, usability problem

Transcript:

Hey, I started filling the master data by seeing values from the old world. I figured that the validation that I put to allow editing the topics is basically not working fine because I disabled it for all instances. I just need to disable it when it's in the view mode, so I will enable it again and then commit my code again otherwise it won't work properly. I thought that I had done the thing correctly but guess what I missed something. [27]

Task in hand: Filling master data

Feelings:

Recording: While filling the master data

Inductive codes: accidental clicks

Transcript:

One problem that I have noticed is that I accidentally keep clicking the save button under the table instead of clicking new so it's not placed at the best spot but I don't know what else I can do for that so I'm going to leave it as it is so but I think this is something that I need to take care of from the usability standpoint. [28]

Task in hand: sanity testing the deployed application

Feelings: Accomplished, confused, frustrated, bored

Recording: While testing the deployed application

Inductive codes: deployed, data entry again

Transcript:

So, the application has been successfully loaded on the free cloud and I am able to login using the demo credentials, but I saw that all the information that I had to put in the file on my local host is not available here. It kind of makes sense. It's going to be a bit of re-work as I'll have to enter the master data again but once I enter entire data on the deployed application, it should be fine. A little frustrating and monotonous because I have to do it again, but this is hopefully the last time that I'm doing this data entry. [29]

Task in hand: Complete the navigation

Feelings: Confident, satisfied, happy

Recording: in the middle of navigation flow, starting css

Inductive codes: usability increase, buttons. css, fun

Transcript:

All right. So, this morning I forgot to start with a voice note but have been in this session for quite some time. What I've been trying to do is add the navigation flow, the log out buttons and stuff like that. Plus, as it has been a long development cycle that I have gone through, so I figured that I would probably start doing some CSS. Going to fix some spacing, do the headers, add some icons and stuff that makes it look better than it did. The TECHFIT logo. So, it's the kind of stuff that I have always liked so, I'm actually enjoying the whole thing. It is a little tough because I'm doing it for the first time but just trying to figure it out by hit trial is actually fun. So far, I have am halfway through I will hopefully be done in another one hour and then I can check in and apply the changes. [30]

Task in hand: Completing bits of navigation and adding CSS

Feelings: Amused, confident

Recording: in the middle of making changes

Inductive codes: jumping between tasks and pages

Transcript:

One of the interesting things that I have realized is that you cannot stay in a single story. You make some change and then realize that something in some other module also needs to change. Also, I have not been very consistent with code commits. I forget to commit small chunks and just end up doing big commits. The good part is that I am the only one doing this, so I don't have to think about merge conflicts etc. [31]

Task in hand: Update what's missing and what's odd

Feelings:

Recording: After committing minor fixes

Inductive codes: commit,logout,image linking,pretty pages

Transcript:

So, update on the status. I have made another commit and this commit consists of the changes related to the logout button. so basically, I had missed the logout button a few places and two of the pages were missing the headers. So, I think it's just the point where you get used to seeing certain things and you just assume that they are there, and you don't take a good look at it. But after taking a break, I was able to see that I had missed a few places. So, I went ahead and did that. [32]

Then another usability issue that I found was that admin dashboard has the four different images for each tile, but these images did not go to their respective pages. So, it is a small change to map the hyperlinks to go to the respective pages. So now if you intuitively just click on the image it would still go to that particular page. So that's also done. The next thing that I'm going to do is probably make the questionnaire page look pretty. Right now, it's just a bunch of text boxes. It looks really weird and really bad. So, I need to figure out a way to show it in a good clean way. [33]

Task in hand: Improve look and feel

Feelings: Confident, happy, satisfied, accomplished

Recording: After remodeling the look of the application

Inductive codes: pretty pages, same old CSS, easy manipulation

Transcript:

So, this one was quick I have been making a lot of progress on the look and feel of the application I've changed a lot of pages. I have made the header look consistent. I tried it on the mobile app for the first time and it looks OK. In my mind, I know that few changes needed to be done for the mobile version of the application. So, the good thing about this tool is that the usability standpoint it is great as it is letting me use the same developer tools and the same CSS classes that I used to use in my day-to-day development back in old Nokia days. So, it's not something that you have to learn again. [34]

Then the other thing that I noticed is the moment you want to change something from say a text box to label, want to make something editable or not editable or restrict showing the label for a particular entity or not it's all just one click away. So that has made a lot of impact in the amount of work that I can get done in a short time. So, I think it's all about knowing what all options are available to you and then using them wisely. This is probably the one of the high-power times of my app development. This is probably the one of those high times of my application development where I am feeling confident about this application and also feeling confident that Prof. Alka would be satisfied with this new application. that's it. [35]

Task in hand: Do something about the questionnaire page

Feelings: Thinking, searching, researching, trying, happy, excited

Recording: After commit

Inductive codes: pretty pages, new fancy thing, deploy feedback

Transcript:

So, I made another commit. Made quite a large number of changes today due to little things that I am finding every time I go into the app. One thing that I figured was that there was a lot of white space in the topics in the rubric table so, I basically left aligned the second column. It looks a lot better than before and is readable now. So, I think he left aligning it worked pretty well. [36]

So basically, today has been more about the look and feel of things and a little about functionality. The other thing that I have done is that I added a group box for the questionnaire which allows me to use all the questions as an accordion, which is letting me close and open each section as per the topics. So, I think that would be useful for the users. and hey it took me like probably five minutes to configure it to a group box and change the functionality to map to it which was quite cool because last time I checked if I was doing this on bootstrap this would have taken me quite some time and lot of iterations. [37]

I am going to deploy the application again and send out an email to Prof. Alka telling her about the changes that of been done. I saw that she hasn't logged in yet so hopefully it was for the good because it made it more user friendly and pretty. So, let's see what she says. [38]

Phase 2 begins

Task in hand: Complete the logic for answered questions

Feelings: Relaxed

Recording: Before starting the dev session

Inductive codes: none

Transcript:

So, it's been 24 hours without looking at the modeler. I am going to start looking at the logic that I had left incomplete yesterday. check if that works and also make CSS edits to the placeholders I had added yesterday. [39]

Task in hand:

Feelings:

Recording: After dev session and commit

Inductive codes: work on feedback

Transcript:

Phew. changes have been committed. I can barely stay awake as it is too late. I guess with this development that had started yesterday. I think it's working today so I have committed my changes and will send an update to her. I have been getting through all the feedback provided by her and making constant progress. I ll wait to see what she thinks about the changes and then work on any additional feedback that comes in. [40]

Task in hand:

Feelings:

Recording:

Inductive codes: feedback,data grid add on

Transcript:

So I've been working this morning to basically complete the changes that were part of the feedback that professor like I had given me and it is a small little changes in the weirdest pages in trying to get things ready for us for the feedback one other thing that I have and I figure it is that I need David to basically store whether a particular dad has it he did a group or not so I'll probably make some more changes in the domain model are the Boolean or something to you no note to whether the group has been dated or not and then you know take things from there so I little worried because of the timeline of the time at the working also for my part-time job but I think I'll figure this out. [41]

Task in hand: Get the back button right

Feelings: Productive, happy

Recording: After setting the back button right

Inductive codes: usability issue solved, pretty pages

Transcript:

So what's new! well... i got some real work done in this session. I have been thinking about the whole back button on the left side issue. it was just messing with my mind for such a long time. So, finally figured it out, the tables were taking so much space and it wasn't really required. So, I decided to use the space on its left once I reduced the width of the tables. a few CSS changes and it looks sweet! [42]

So, all pages with the tables are now in the new layout. Plus, I color coded the action buttons and made them consistent. I guess styling the application is what I like the most. Cool. overall a productive and happy session!! [43]

Task in hand: Complete the judge edit logic, update status

Feelings: Happy, troubled, irritated

Recording: After pushing judge edit changes

Inductive codes: tough feature development

Transcript:

Recording this a little late because I forgot to record this after completing my commit because I had to go out but basically in the last session I continued on what I was already working on edit new thing for the judge. I am almost done with it I have split some pages and figure it out. So the next step tomorrow is to take care of the three remaining items that are open for development and then update the sprinter and send out an email to Prof. Alka that the whole application is ready for review so that's about it. Overall I have made a lot of progress today and I am happy and satisfied in that respect it's just that this issue that I've been facing has been troubling me and basically irritated me a little. [44]

Task in hand: Location logic needs to be figured out

Feelings: Neutral, ideation

Recording: Before starting the dev session

Inductive codes: tough feature development

Transcript:

So, I sent out an email to prof. Alka about everything that is done. Just three things remain. the location integration, question validation check and start/edit handling for judges. I have been delaying looking in the location issue as it did not work last time. but I think it's part of the main functionality so its high time that I look into it and make it work... the current thought is too search for a possible add on that will allow me to add location to the drop down on the fly. If that does not work, I will try to have something less fancier. Let's see how this goes. [45]

Task in hand: Commit the location integration and deploy the application

Feelings: Very happy, accomplished, relieved

Recording: after pushing the final commit for phase 2

Inductive codes: location integrated, tough feature development, testing

Transcript:

Finally! Yes, so I am done with the location integration. I tried to look up for a fancy dropdown but after wasting about half an hour in figuring that out, I gave up. I moved to my back up option to add a button for location on group page. Had to make changes to domain model to accommodate the new entity and get rid of the current location attribute. Then updated all pages that were using the location attribute. minor hit and trial and small adjustments. It started behaving the way it should. I checked all the pages and went through one demo run and it works fine. [46]

This brings me to completion of all functionality required for the application. The only thing that remains is minor bug fixes and some good to have things in my mind. Will take care of that later. This is a big moment for me. I am happy. Hopefully will get some positive feedback from Prof. Alka on this too!! [47]

End of phase 2

APPENDIX B. APPROVED IRB PROTOCOL

PROPEL 01030689



HUMAN RESEARCH PROTECTION PROGRAM
INSTITUTIONAL REVIEW BOARDS

To: Alka Harriger

From: Purdue University Human Research Protections Program (HRPP)

Title: Using Rapid Application Development for University Software Development Projects

Date: 2019-01-28

Re: Exemption Determination

Through the answers you provided in response to questions in the Purdue Research Online Portal Exemption Logic (PROPEL), Purdue's HRPP has determined that the research project identified above qualifies as exempt from IRB review, under federal human subjects research regulations Exemption Category 2/3 [-Anonymous Educational Test, Surveys, Interviews, or Observations of Public Behavior; e.g., 45 CFR 46.101(b)(2) or (3)].

The answers provided in PROPEL indicate your plans to:

- Conduct anonymous educational tests, surveys or observation of public behavior that do not place participants at risk of criminal/civil liability or damage to their financial standing, employability or reputation.
- Involve only participants over the age of 18.
- Indicate to each participant that the research is voluntary any questions may be skipped.

Figure B.1. IRB exempt status

APPENDIX C. RESPONSES TO OPEN-ENDED SURVEY QUESTIONS

A - Briefly describe up to the three most negative aspect(s) of using Mendix

1. Overwhelming amount of boxes on all screens
2. Some key aspects require RD attention such as performance, document generation, language support
3. All data has to be modeled (can't have "loose variables" for UI)
4. sometimes run into un-explained behavior - mendix generates divs that has a class that changes on click automatically. Then you may have to work around this
5. Other developers don't see it as useful, so it doesn't get the "help" it needs
6. Integrating more complex actions takes getting used to
7. Random Bugs that need to get fixed from new releases of the product
8. needs expertise for high level performance
9. calculated attributes are slow and unoptimized.
10. Frequency of updates makes long term support problematic
11. Very hard to create very secure applications
12. Learning Curve High
13. It's missing some QoL features like auto-fill CSS and split-screen
14. Simple things you would do in code are less easy to do in Mendix
15. Constant new versions
16. Current releases have a tendency to be a bit buggy
17. The document generator is so far out of date it's barely usable
18. Price
19. Sometimes I need to click to much to achieve simple things
20. Doesn't feel like a native application.
21. Quite a lot of fixes in every new Mendix-release
22. Build time super slow
23. Poor data grid features and rich text editor lacks images
24. no wysiwyg at all
25. Support isn't always taking your requests seriously until some high ranked individual steps in
26. If apps require tons of custom code, then mendix may not be the best use for these scenarios
27. When working on multiple projects with several iterations of the Modeler required, it can take up a good chunk of disk space

28. Limits on globe variables
29. limited document generation possibilities
30. inline style don't appear as drop-downs like browser dev tool has. So styling is pain to remember when doing simple things like aligning or sizing or coloring.
31. Requires a fairly high level of CS knowledge
32. Document generation (out of the box) is weak
33. Require more Video tutorials
34. its missing some random implementation features like the ability to search a data grid by association/ microflow
35. UI isn't as flexible as it should
36. NO friendly css interface
37. Mendix themselves think paid alternatives are a viable alternative to an inhouse document generator
38. Document templates
39. Free cloud sometimes borks
40. Creates a lot of code under the hood that's unnecessary
41. Re-usability when more then one Mendix app is used needs extra effort to accomplish
42. Lacks normal programming things
43. Inability to commit only selected changes
44. expensive
45. Mendix expert services often lacks actual expertise for expert advise
46. Forced to use windows OS
47. Free app doesn't use a conventional database
48. expensive for single application licenses micro-services support is absent. Deployment granularity is absent. Licensing plan is primitive. No support for NoSql db. No support for module level db. Integrating with existing db tables is pain. I could go on like this all night.
49. Very hard to create custom widgets
50. Need improvement in APPS testing
51. it's a bit pricey in their cloud license, so it's not for smaller companies viable
52. Document editor isn't sufficient
53. Hard to insert custom code
54. Performance in a multi tenant environment is not as good as it should be
55. Expensive/Not Open Source
56. Can be difficult to decide which widgets to use because of lack of info and support

B - Briefly describe up to the the three most positive aspect(s) of using Mendix

1. i can work with other team-members
2. Easy to start with the platform
3. Entirely self-contained (database, server logic, UI/UI logic)
4. Ability to get Proof of Concepts/MVPs out very quickly
5. Most complex issues can be solved in a way that is easy for others to review and use again
6. Quick and easy to hit the ground running when starting a new project
7. Quick and easy to use
8. easy to use and learn
9. Nanoflows
10. Can equip a process oriented person with a tool that enables the development of apps
11. Very easy to create fairly robust applications
12. Less hassle for simple Apps
13. You work on the data structure and the page design at the same time
14. Easy to learn
15. easy to use and deploy
16. Easy deployment
17. it's quite easy and quick to update your application
18. Ease of use
19. Most of the times I can achieve what I want, with limited clicks
20. Can develop an application extremely fast
21. For an experienced developer, you're able to develop fast
22. Enables very high productivity
23. Easy to build sophisticated applications
24. Extensible through Java/JavaScript for cases where out-of-the-box is insufficient (which are few)
25. Business logic is usually easy to implement
26. Flexibility between desktop modeler and WYSIWYG modeler
27. Can modify changes in front of the client.
28. easy integration
29. java integration
30. Very easy to integrate with other systems
31. Support is good

32. its easy to hand an application to users for feedback
33. Don't need to worry about setting up the non-trivial things in your typical project
34. fast
35. Easy cloud-based work-space for project management (agile)
36. you don't need to know a lot of coding to use it
37. The oil between different IT systems
38. I can iterate quickly over ideas myself and with the customer
39. Easy to change code later on
40. Low code
41. Changing names and finding usages project-wide
42. A lot of basic needs have been properly implemented which lets you focus on your process/application instead of the things like authentication.
43. More complex tasks are still possible by extending platform with custom java/JavaScript
44. Revision changes and backups easily accessible
45. quick results vs low effort
46. external library addition
47. Very easy to deploy applications
48. its code is easily visualizable
49. It's fun to see people getting excited for app development, the "I can do that" mentality
50. effective
51. It's just so far to build software
52. API's
53. A lot of help can be found in the community
54. A lot of great tools like build in version control, a cloud server and debugging tools.
55. Smaller gap between developer and product owner (visual representation of code)
56. Ease and consistency of deployment

APPENDIX D. OTHER AUTOETHNOGRAPHIC DATA

Personal reflection 1

Q1. What were the Biggest incentives in doing this?

SA: To not get worried about petty little things like logins and not worry about putting everything in place and then hosting it on a server and taking care of connections and stuff like that. For small applications like this, it just makes sense to do something like that. Take a rapid application development tool where you can typically just configure the whole thing and it becomes a fully functional application easily.

Q2. If u were to say it was easier or more difficult to do CSS in Medix?

SA: I think it's all about getting used to it. I was playing with it yesterday for the first time, trying to learn to do CSS. First 5 minutes were spent on figuring out how to do it but once you see that it works the same way as you would do normally, you have the same CSS styling. you just need to figure out which div you need to put it in. So, I would say it's easy.

Q3. Which part do you think was the toughest?

SA: I think because I had a prototype application for analyzing and seeing what all needs to be built and then figuring out the most apt way to do so. I had to change my logic a lot of times but how would you know which field to update unless you start developing the application and reach that point.

Old application has a lot of extra clicks you literally have to select everything and then click next to go to the next page. So, I wanted to keep the structure similar to what was in the old applications so that anybody who has used the old application can easily switch to the new application. But increase the usability by making it better and using tables instead of all those weird columns.

Q4. So, lets talk about the high points and the low point of this whole process?

SA: High Point was when I was able to see the flow go through for the first time. The second High Point was yesterday when I was fixing the CSS and what was happening there was that I was able to fix all the pages very quickly. Just imagining that I wouldnt have to manually program this if this did not work and would make my life so much better.

The low point was trying to get figure out how to put the name of the user on each page. Figuring out that I needed a microflow for it and put it in that stupid I never used it is a very simple fix and missed one yes no statement and that is what I figured you have stories you have eaten a few more do you land you want to work on a single story of the time but for me I just printed right now everything is in running State. It's not like you were doing user management and you know you need to make that change in other stories but won't do it until you start a new story. If you know that this needs to happen on all the pages, you will go ahead and do it.

Q5. What could you improve if you had to build another application?

SA: I think I would take a lot less time to start the application because this was the first time that I started an application from scratch with new things. what else could improve would be that I did not have the stories listed in the beginning because I thought I'm the only person I know what all needs to be done and I can just keep doing it but it's always good to have all the stories listed because then you know what all is in progress at any given time. I would also check-in at regular intervals as doing it after one full day of development, you have like 35 files that you need to commit.

Personal reflection 2

Q1. So what happened after you finished Phase 1?

SA: Finishing phase 1 was when I basically sent out the credentials to be a professor for testing the application and she tried to go to each and every module and give me some feedback on items such as some screens were not closing on clicking the back/cancel button. Another one was where she was lost in between the transmission of a few pages. So the focus of phase 2 is building the application based on the feedback and the remaining tasks.

Q2. How did she describe the things that you need to do for Phase 2?

SA: I tried to suggest using the feedback tool but she basically just wrote everything down in an email. Telling me what she did and what the problem that was encountered. I have to do add the feedback manually into the tool.

Q3. Would you say that things that she wanted were things you would have caught if you looked harder?

SA: There were a few things that I already knew were broken and I was in the process of fixing them or some good to have things like the back button. So, I was kind of in the middle of it but if you think that I would have not identified it, it would be wrong. The thing that I would not have caught was the request to keep the years going till later rather than just 2020. I am sure, I would not have thought about that one and might have left it as is.

Q4: Did she make any recommendations for any look and feel related things?

SA: Yeah, she did make some recommendations for the back buttons, for table spacing and other stuff. So I started working on those keeping in mind my own feedback and her feedback together.

Memos

1. The whole login page thing was so nice... Thank god I did not have to write weird reg-ex codes to ensure that the password policy was sound. It was such a mess in my last project.

2. I keep saying microflow and domain model which is like a common term in Mendix but no one outside it would understand, I need to make sure that I explain it well and in simple words.
3. The code commit thing does not look important here as I was the only one building this, but i need to mention what happens when 3 people are trying to work on the same thing.
4. This whole managing the time between research and other things got over my head to the point where I mentioned it in one of the voice notes which looks so out of context!
5. I need to emphasis on the importance of feedback and the role of user in shaping an application.
6. Interview 2 was based too much on what kind of feedback came in and how I handled it, should have kept it more structured rather than open.

APPENDIX E. APPLICATION SCREENSHOTS

The following pages provide visual comparisons of user screens of the current ASP.Net judging and the researcher-developed Mendix app.

Login:

TECHFIT - Login

Please enter your user name and password to gain access to secured parts of this website.

Log in to TECHFIT

Username:

Password:

You must log in to view judging pages

[Brochures](#)
[Contact](#)

This project is supported by the National Science Foundation, awards DRL-1312215 and DRL-1640178. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Figure E.1. Old login screen

Sign in

User name

Password

Figure E.2. New login screen

Admin dashboard Screen:



Figure E.3. Old admin dashboard

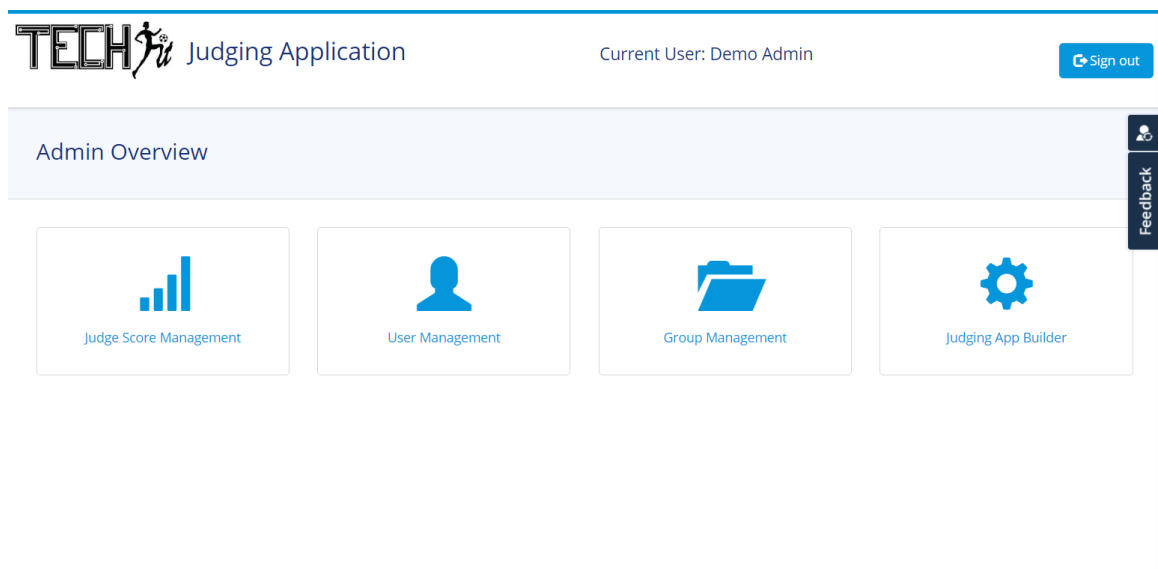
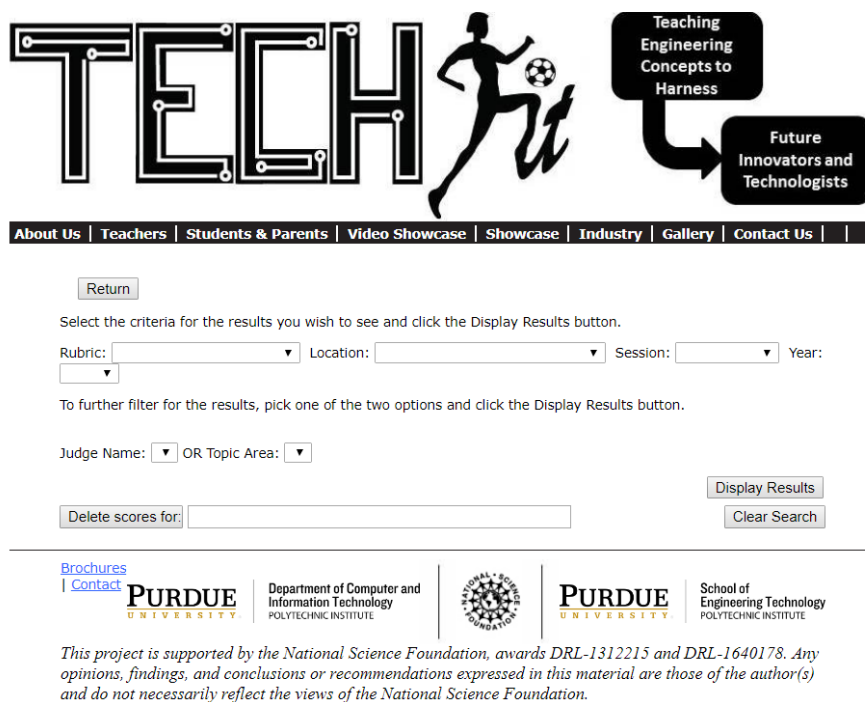


Figure E.4. New admin dashboard

Score management screen:

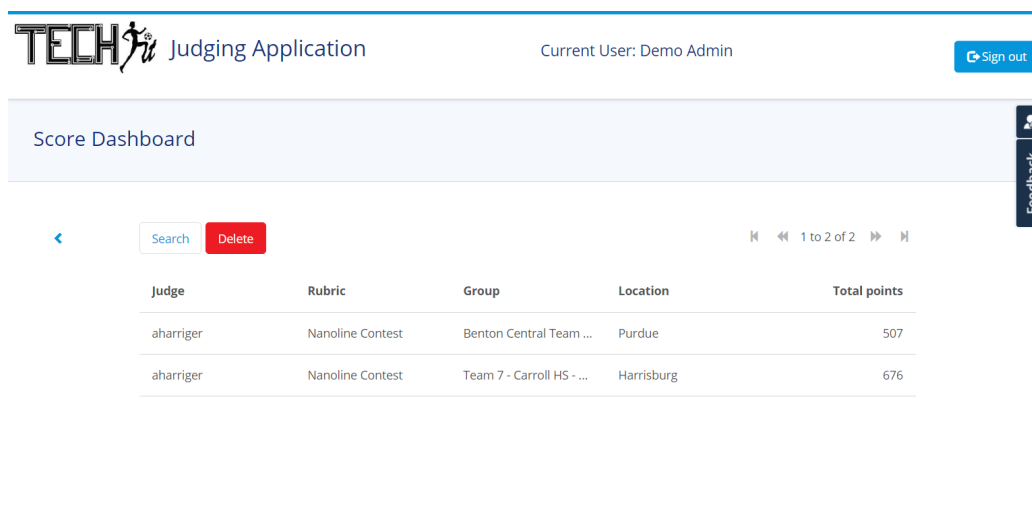


The interface features a header with the word "TECH" in a stylized font, followed by a silhouette of a person running with a soccer ball. To the right, two callout boxes read "Teaching Engineering Concepts to Harness" and "Future Innovators and Technologists". Below this is a navigation bar with links: About Us, Teachers, Students & Parents, Video Showcase, Showcase, Industry, Gallery, Contact Us, and a vertical bar.

Below the navigation bar is a "Return" button. A text prompt says: "Select the criteria for the results you wish to see and click the Display Results button." Below this are four dropdown menus: Rubric, Location, Session, and Year. A text prompt follows: "To further filter for the results, pick one of the two options and click the Display Results button." Below this are two more dropdown menus: Judge Name and OR Topic Area. At the bottom left is a "Delete scores for:" label followed by a text input field. At the bottom right are two buttons: "Display Results" and "Clear Search".

At the bottom of the page, there are logos for Purdue University, the Department of Computer and Information Technology Polytechnic Institute, the National Science Foundation, and the School of Engineering Technology Polytechnic Institute. A disclaimer text reads: "This project is supported by the National Science Foundation, awards DRL-1312215 and DRL-1640178. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation."

Figure E.5. Old score management page

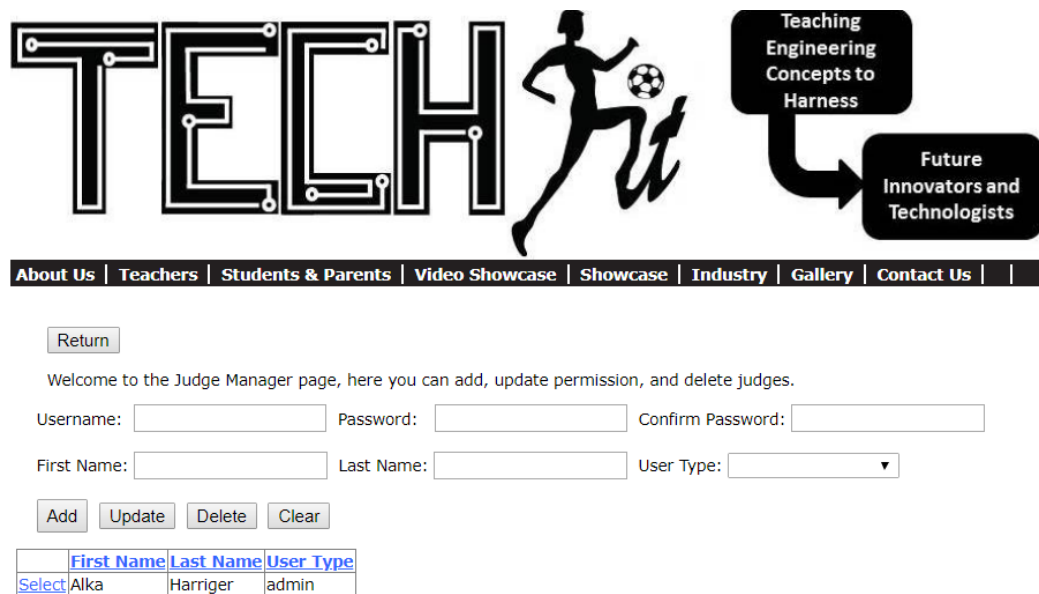


The interface has a header with the word "TECH" in a stylized font, followed by "Judging Application". To the right, it says "Current User: Demo Admin" and a "Sign out" button. Below the header is a "Score Dashboard" section. On the right side of the dashboard is a vertical "Feedback" button. Below the dashboard is a table with columns: Judge, Rubric, Group, Location, and Total points. The table contains two rows of data. Above the table are "Search" and "Delete" buttons. To the right of the table is a pagination control showing "1 to 2 of 2".

Judge	Rubric	Group	Location	Total points
aharriger	Nanoline Contest	Benton Central Team ...	Purdue	507
aharriger	Nanoline Contest	Team 7 - Carroll HS - ...	Harrisburg	676

Figure E.6. New score management page

User management screen:



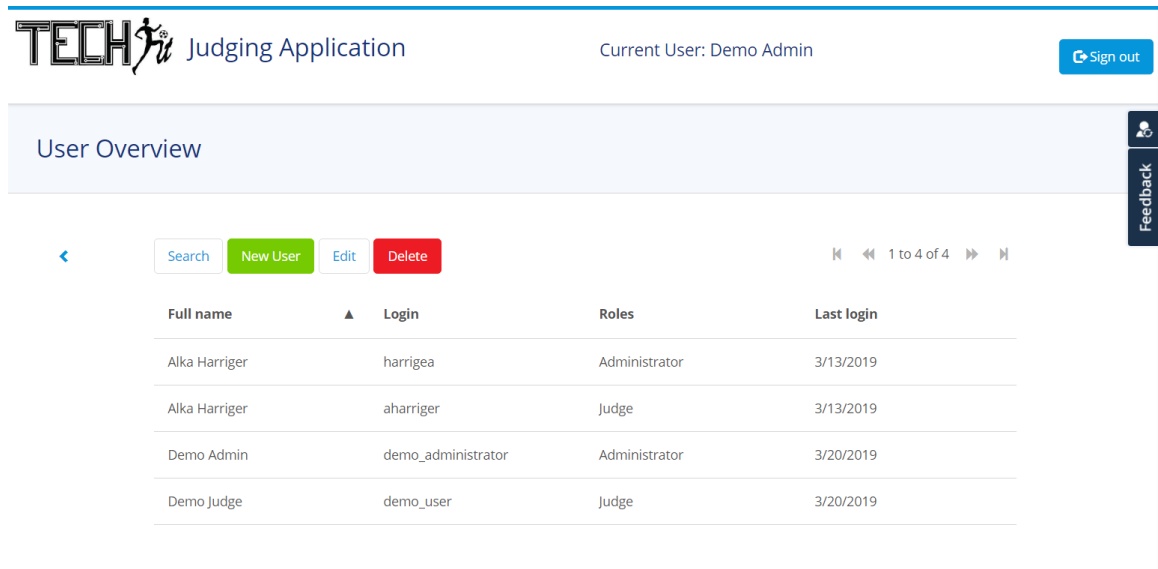
The interface features a header with the word "TECH" in a stylized, circuit-like font, followed by a silhouette of a person kicking a soccer ball. To the right, two callout boxes contain the text "Teaching Engineering Concepts to Harness" and "Future Innovators and Technologists". Below the header is a navigation bar with links: About Us | Teachers | Students & Parents | Video Showcase | Showcase | Industry | Gallery | Contact Us | |.

Below the navigation bar is a "Return" button. A welcome message reads: "Welcome to the Judge Manager page, here you can add, update permission, and delete judges." Below this are form fields for "Username:", "Password:", "Confirm Password:", "First Name:", "Last Name:", and "User Type:" (a dropdown menu). Below the form fields are buttons for "Add", "Update", "Delete", and "Clear".

Below the buttons is a table with the following data:

	First Name	Last Name	User Type
Select	Alka	Harriger	admin

Figure E.7. Old user management page




The interface features a header with the word "TECH" in a stylized font, followed by "Judging Application". To the right, it says "Current User: Demo Admin" and has a "Sign out" button. Below the header is a "User Overview" section. Below this is a table with the following data:

Full name	▲	Login	Roles	Last login
Alka Harriger		harrigea	Administrator	3/13/2019
Alka Harriger		aharriger	Judge	3/13/2019
Demo Admin		demo_administrator	Administrator	3/20/2019
Demo Judge		demo_user	Judge	3/20/2019

Below the table is a "Feedback" button. The interface also includes a search bar, a "New User" button, and a "Delete" button. A pagination bar shows "1 to 4 of 4".

Figure E.8. New user management page

Group management screen:



[About Us](#) | [Teachers](#) | [Students & Parents](#) | [Video Showcase](#) | [Showcase](#) | [Industry](#) | [Gallery](#) | [Contact Us](#) |

[Return](#)

Welcome to the Group Manager page, here you can add, update, and delete groups.

Group Name:


Rubric Associated: Year: Session:

Event attending:

[Add](#) [Update](#) [Delete](#) [Clear](#)

	Group Name	Rubric Associated	Event State	Session Attending	Year Attending
Select	Hauser Showcase	TECHFIT Judging Rubric	Hope, IN	Spring	2017
Select	Winamac Showcase	TECHFIT Judging Rubric	Winamac, IN	Spring	2017
Select	CofC 2017 - Greenbriar-Brenda+Nicole	Teacher Showcase	CofC Summer PD	Summer	2017

Figure E.9. Old group management page



Judging Application

Current User: Demo Admin [Sign out](#)

Group Overview

[Search](#) [New](#) [Edit](#) [Delete](#) [Add Location](#)

1 to 10 of 10

Group Name	Contest	Location	Session	Year
Porter County Career...	Nanoline Contest	Purdue	Spring	2018
Benton Central Team...	Nanoline Contest	Purdue	Spring	2018
Carroll High School - ...	Nanoline Contest	Purdue	Spring	2018
Elizabethtown Area ...	Nanoline Contest	Purdue	Spring	2018
Cumberland Valley Hi...	Nanoline Contest	Purdue	Spring	2018
Team 1 - Dauphin CT ...	Nanoline Contest	Harrisburg	Spring	2019
Team 2 - Elizabethto...	Nanoline Contest	Harrisburg	Spring	2019
Team 3 - Walker Care...	Nanoline Contest	Harrisburg	Spring	2019

Figure E.10. New group management page

Judge app builder screens:

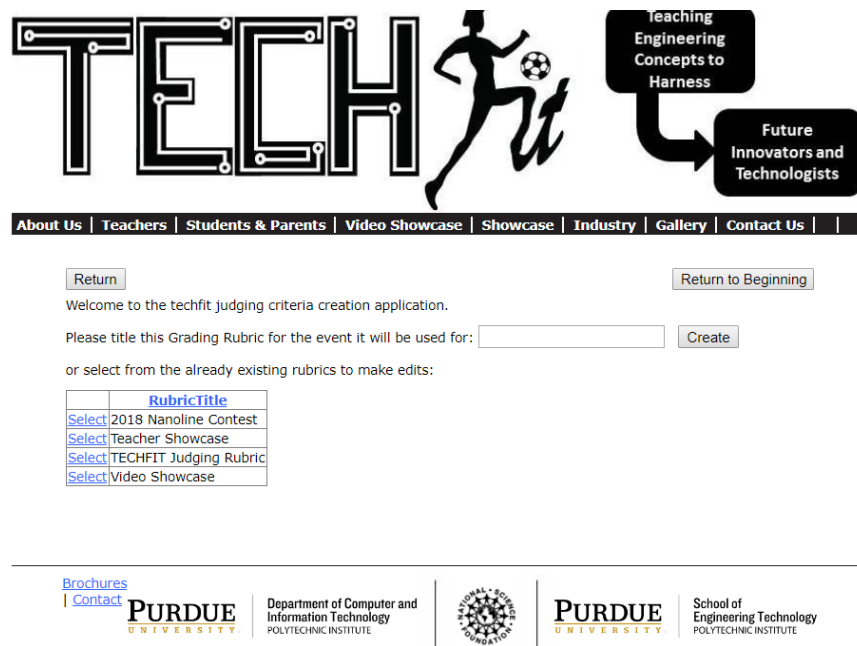


Figure E.11. Old rubric management page

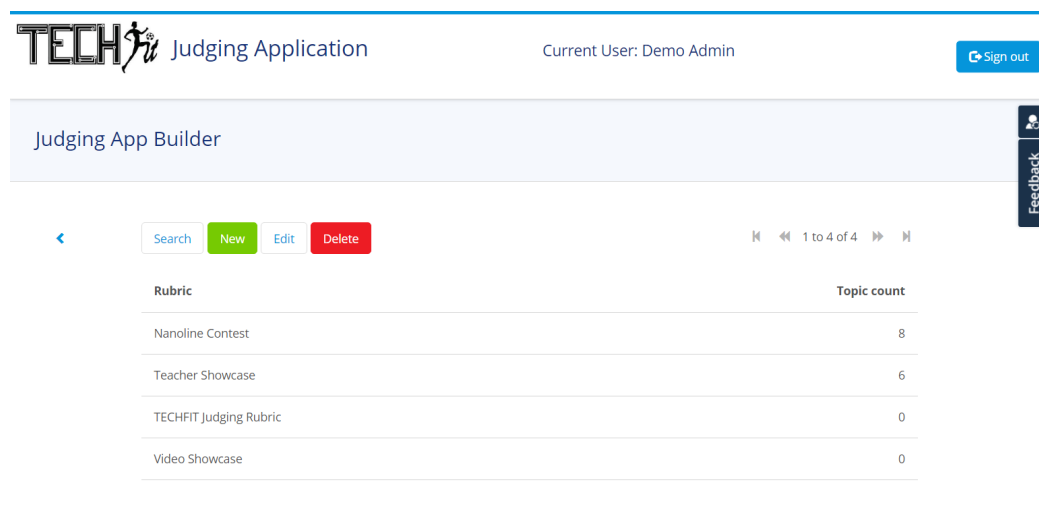



Figure E.12. New rubric management page



[About Us](#) | [Teachers](#) | [Students & Parents](#) | [Video Showcase](#) | [Showcase](#) | [Industry](#) | [Gallery](#) | [Contact Us](#) |

[Return](#) [Return to Beginning](#)

Welcome to the techfit judging criteria creation application.

Please add, delete, or update the topic areas you wish to cover up to 10 topic areas. Hit next when you are done.

[Add](#) [Delete](#) [Update](#)

	TopicArea
Select	Project Concept
Select	Use of Nanoline
Select	Engineering
Select	Craftsmanship
Select	Application
Select	Communication
Select	Teamwork
Select	Entrepreneurial

[Back](#) [Next](#)

[Brochures](#)
[Contact](#)













Figure E.13. Old topic per rubric page



Judging Application

Current User: Demo Admin [Sign out](#)

Judging App builder

Topic	Question count
Project Concept	4
Use of Nanoline	4
Engineering	6
Craftsmanship	5
Application	5
Communication	5
Teamwork	6
Entrepreneurial	4

Figure E.14. New topic per rubric page


[Return](#)
[Return to Beginning](#)

Welcome to the techfit judging criteria creation application.

Please select each topic area then give the points, as well as the text to be added for the question:

	TopicArea	NumberOfQuestions
Select	Project Concept	4
Select	Use of Nanoline	4
Select	Engineering	6
Select	Craftsmanship	5
Select	Application	5
Select	Communication	5
Select	Teamwork	6
Select	Entrepreneurial	4

Topic Area: Project Concept

Question:

Does the project utilize a creative idea as the basis of its concept?

Points:

5

Question:

Does the project display a unique creative strategy in utilizing Phoenix Contact products?

Points:

5

Question:

Does the project have a real meaningful work or play result?

Points:

5

Question:

Can the project be replicated?

Points:

5

Figure E.15. Old questions per topic page

TECH Judging Application

Current User: Demo Admin

[Sign out](#)

Judging App builder
Selected Rubric - Nanoline Contest

Topic title
Project Concept

[New](#) [Edit](#)

1 to 4 of 4

Question	Points
Does the project utilize a creative idea as the basis of its concept?	5
Does the project display a unique creative strategy in utilizing Phoenix Contact produ...	5
Does the project have a real meaningful work or play result?	5
Can the project be replicated?	5

Figure E.16. New questions per topic page

Judge landing page



Figure E.17. Old judge landing page

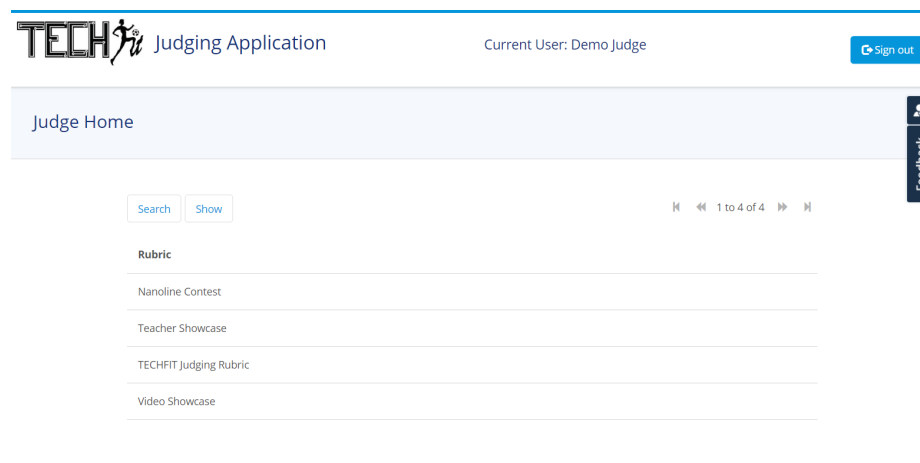



Figure E.18. New judge landing page

Judge group selection page



[About Us](#) | [Teachers](#) | [Students & Parents](#) | [Video Showcase](#) | [Showcase](#) | [Industry](#) | [Gallery](#) | [Contact Us](#) | [Gallery](#)

[Logout](#) [View Results](#)

Location Selection:


Session Selection:

Year Selection:

Group Selection:

[Back](#) [Start](#)

Figure E.19. Old group selection page



Judging Application

Current User: Demo Judge [Sign out](#)

Group Selection
Selected Rubric - Nanoline Contest

[Start](#) [Edit](#)

1 to 6 of 6

Group Name	Scored
Benton Central Team - Soft Contact	Yes
Benton Central Team - Jiggy Jukebox	
Porter County Career Center-ChildProtectCarSeat	
Carroll High School - Auxilia	
Cumberland Valley High School - Nano House	

Feedback

Figure E.20. New group selection page

Judge questionnaire page

TECH

Teaching Engineering Concepts to Harness Future Innovators and Technologists

About Us | Teachers | Students & Parents | Video Showcase | Showcase | Industry | Gallery | Contact Us

Logout

Change Group Submit

Evaluation for **Benton Central Team 1 - Soft Contact** by **Suyash Agrawal**

Current Topic Area: Project Concept

Navigation:

Project Concept Use of Nanoline Engineering Craftsmanship Application Communication

Teamwork Entrepreneurial

Please rate the team on how they completed each of the following categories.

Does the project utilize a creative idea as the basis of its concept?

☐ Missing ☐ Minimal ☐ Acceptable ☐ Good Job ☐ Wow

Does the project display a unique creative strategy in utilizing Phoenix Contact products?

☐ Missing ☐ Minimal ☐ Acceptable ☐ Good Job ☐ Wow

Figure E.21. Old questionnaire page

TECH Judging Application

Current User: Demo Judge

Sign out

Questionnaire

Selected Group- Benton Central Team - Soft Contact

All questions need to be scored before submitting

Back

Project Concept - 4 Questions

Does the project utilize a creative idea as the basis of its concept?

☐ Missing ☐ Minimal ☐ Acceptable ☐ Good Job ☐ Wow

Does the project display a unique creative strategy in utilizing Phoenix Contact products?

☐ Missing ☐ Minimal ☐ Acceptable ☐ Good Job ☐ Wow

Does the project have a real meaningful work or play result?

☐ Missing ☐ Minimal ☐ Acceptable ☐ Good Job ☐ Wow

Feedback

Figure E.22. New questionnaire page