



Universiteit
Leiden
The Netherlands

Opleiding Informatica & Economie

Increasing the Effectiveness of Communicating Release Notes
to Users: A Case Study in Salesbuildr

Claire Leonie Theresia Elke Fietelaars

Supervisors:
Joost Visser & Natalia Amat Lefort

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

24/06/2024

Abstract

This thesis investigates methods to enhance the communication of software release notes to users. Effective communication through release notes is critical in bridging the gap between developers and end-users. The study focuses on improving effectiveness of release notes to facilitate better developer-user communication. This research is conducted at software company Salesbuildr. Employing a Design Science Research Methodology (DSRM), this research develops an artifact aimed at enhancing release note practices. Utilizing a mixed methods approach involving surveys and interviews, the artifact, a virtual walk-through, is designed and implemented. Quantitative and qualitative evaluations indicate increased user engagement and positive feedback. However, limitations in data accuracy and scope hindered the assessment of results.

Contents

| | | |
|-----------|--|-----------|
| 1 | Introduction | 1 |
| 1.1 | Thesis overview | 1 |
| 2 | Background | 2 |
| 2.1 | Release Notes | 2 |
| 2.1.1 | What Should Release Notes Contain? | 2 |
| 2.1.2 | Writing Process of Release Notes | 3 |
| 2.1.3 | Best Practices | 4 |
| 2.2 | Salesbuildr | 5 |
| 2.2.1 | Current Situation at Salesbuildr | 5 |
| 3 | Related Work | 6 |
| 4 | Method | 8 |
| 4.1 | Design Science | 9 |
| 4.2 | Methodological Approach | 10 |
| 5 | Definition of Requirements | 11 |
| 5.1 | Survey | 11 |
| 5.2 | Interviews | 13 |
| 6 | Design and Development | 14 |
| 7 | Evaluation | 15 |
| 7.1 | Quantitative Evaluation | 15 |
| 7.2 | Qualitative Evaluation | 16 |
| 8 | Results | 16 |
| 9 | Discussion | 18 |
| 10 | Conclusions and Future Research | 19 |
| | References | 21 |

1 Introduction

This bachelor thesis, conducted at the Leiden Institute of Advanced Computer Science (LIACS) under guidance of Joost Visser and Natalia Amat Lefort, examines the underexploited potential of software release notes as powerful instrument for user engagement.

In the dynamic world of software development, release notes are the unsung heroes that make software implementation and updates manageable. Despite their importance, many companies struggle to effectively engage users through release notes. This thesis explores the ways in which release notes can be made more effective in reaching and engaging users.

To increase user satisfaction and optimize the adoption of the software, effective communication of the software updates is essential. Inadequate release notes can lead to decrease of user satisfaction, suboptimal use of the software functionalities and an increased demand for support [3]. While existing research has explored various aspects of software documentation, such as automating the production of them[11] or analysing the current content of the release notes [22], there remains a significant gap in understanding and implementing best practices for crafting informative and engaging release notes.

The problem addressed in this thesis is the lack of effective methods for creating release notes that engage and inform users. Current practices can result in lengthy, technical, and unappealing documents that fail to capture user interest or transfer the new information from the release notes efficiently. This study, centered on Salesbuildr, a company dedicated to streamlining user workflows, seeks to bridge this gap. The findings and methods developed here are not only intended to benefit Salesbuildr but also to provide valuable insights for other companies in the industry.

This research aims to develop and evaluate ways for improving the effectiveness of software release notes. The specific objectives are to analysing current practices in release notes creation and communication, identify the best practices, and develop an artifact based on these insights and evaluate its impact on user engagement and satisfaction.

To address these objectives, a design science research methodology will be utilized. This approach involves an iterative development and evaluation of designed artifact to solve the identified problem. Data will be gathered through surveys, interviews, and user feedback to design the artifact and assess the effectiveness of the it. By achieving these objectives, this thesis aims to advance how release notes are crafted and communicated, turning them into powerful tools for enhancing user engagement and satisfaction.

1.1 Thesis overview

The thesis is structured as follows: This chapter contains the introduction 1 and provides an overview of the topic, context and objectives. Section 2 provides knowledge and terminology used in this thesis. Section 3 reviews the existing research conducted on relevant topics for this study. The methodology used to conduct this research is described in section 4. Section 5 describes the way the requirements for the artifact were gathered. in Section 6 the development of the artifact is described. Section 7 then evaluates the artifact. The findings of the research are presented in Section 8. The limitations and implications of the research are discussed in Section 9. Finally, section 10 summarizes the research and proposes some future research topics.

2 Background

Effective and clear communication is essential in the software development industry, particularly when informing users about updates and changes. This chapter offers background knowledge and essential terminology related to release notes and their communication, which are critical documents that accompany software updates. Understanding these definitions is fundamental to comprehending the importance of release notes and the processes involved in their creation and dissemination.

Firstly, a definition of release notes is given along with their purpose and importance. Secondly, The essential components of a good release note are given. Including these should help make the release notes more comprehensive and useful for users. After this, The writing process of the release notes is discussed. Finally, a set of best practices will be discussed for creating both efficient and accessible release notes, maximizing their effectiveness and enhancing user experience.

In addition, this chapter introduces Salesbuildr, the company where this research is conducted. Understanding the context of the company is useful for understanding the specific challenges and opportunities encountered in the communication of release notes.

2.1 Release Notes

Release notes are documents accompanying software updates, offering users details on changes, improvements, and fixes in the latest version of the software. Serving as a communication channel between the development team and users, release notes play a pivotal role in informing users, facilitating smooth software transitions, and aiding in troubleshooting [4].

These documents typically detail new features, modifications, and resolved issues [22]. Release notes are designed to deliver vital information to users, customers, and stakeholders, ensuring they remain engaged and well-informed about the software's development [23]. By demonstrating ongoing improvements and responsiveness to user feedback, release notes can enhance user satisfaction and build trust in the product, fostering transparency by addressing current software issues [5]. Moreover, release notes manage user expectations, preparing users for updates and changes. Additionally, well-crafted release notes can boost adoption rates [18], potentially reducing the need for extensive user support.

2.1.1 What Should Release Notes Contain?

Release notes should provide detailed information about the changes introduced in the update, enabling users to grasp the impact of each release. The content of release notes depends on the specific project and cannot be universally applied to the entire software [13]. For this study, the focus is on product release notes of regular releases. Release notes for pre-releases, beta or alpha releases are not considered because they are often not included in release notes or very short and lacking information [14].

Through analysing various sources on release notes, the following components were identified as essential for creating informative documentation for users. Sources consulted include: [16], [18], [3], [17], [2]:

1. **Title and version information**

Give the user with a clear understanding of the current release. This section can include a title, release date and version number of the release.

2. **Clear and concise summary**

Provide an overview that highlights the main changes in the release.

3. **New features**

state new features or enhancements of new features. Each new feature should have a feature description and usage instruction. The description explains the user what the new feature is, why it was added and what benefits it could have. The usages instructions help the the reader to use the new feature themselves.

4. **Bug fixes and improvements**

List the resolved issues and state the performance enhancements.

5. **Known issues**

Acknowledge ongoing issues to the users and inform if there are workarounds for this issue.

6. **Technical details**

Provide information on technical requirements such as compatibility notes, system requirements or installation instructions.

7. **Extra resources and support information**

Offer further assistance by linking information such as the user guide or support contact information.

Release notes can take various forms, utilizing different templates, but their primary goal remains consistent: to provide relevant information to the target audience[17]. For example, software release notes often focus on the technical aspects of an update by providing a detailed changelog, whereas product release notes emphasize new features and their applications. Identifying the intended audience is essential before drafting the release notes[5].

While release notes do not have a defined structure, there are common practices that many companies follow. Amoeboids [3] state that release notes typically include the following elements: a header, an overview, issue summary, resolutions, and impacts. This structured approach can help ensure that the information is organized and accessible to the reader. For this research, the focus will be on product release notes, as the aim is to investigate effective communication methods to end-users.

2.1.2 **Writing Process of Release Notes**

Creating effective release notes involves several key steps: planning, gathering information, drafting, reviewing, and distributing. Each step can help ensure that the release notes are accurate and user-friendly.

The planning stage is crucial for ensuring that the release notes are comprehensive, accurate, and user-focused [2]. Identifying the target audience is a fundamental part of this step, as it helps determine the level of technical detail and tone. Different audiences, such as end-users, developers, and investors, have varying needs and interests [9]. End-users want to understand how changes affect their use of the software, while developers require detailed technical information, and investors are interested in the overall development progress.

The gathering of relevant information involves collaboration with development and product management teams to fully understand the changes and improvements made. The development team provides technical details, while the product management team highlights key updates that align with business goals [16]. Clearly defining the goals of the release notes is essential. These goals can be to encourage users to interact with new features, explain changes, and establish trust by addressing known issues and details on how to fix these.

Determining the right writer is essential as they need to have in-depth knowledge of the software and excellent communication skills to translate this knowledge into clear, readable language [4]. The choice of writer may vary depending on the targeted audience. For example, technical writers might be ideal for developer-focused release notes, while marketing or customer support teams might contribute to notes aimed at end-users. Amoeboids [3] suggests forming a team to draft release notes, incorporating input different departments such as customer support or marketing to ensure the content is engaging and visually appealing.

The drafting process begins with creating an initial version of the release notes. The information can be organised logically by using a consistent format, using headings and subheadings [2]. This can help make the information more digestible for the readers. Writing in clear and concise language tries to ensure that each section is to the point, only providing the necessary information to the reader without overwhelming them.

After completing the draft, the reviewing process starts. Here the documents gets reviewed to detect errors or unclear sections. After gathering the feedback and adjusting the document accordingly, a proofreading will check for grammatical errors and formatting the release notes consistently [4]. This can help ensure that the document looks polished and professional. A final approval, before distributing the document, can be done by the key stakeholders, ensuring that all essential information is included and properly presented.

After the writing process of the release notes is complete, the document gets distributed to the intended audience. This can be done through multiple channels such as email, website, or in-app notifications. Choosing the right channel is important to reach the user effectively [5]. The timing of publishing the release notes is important, ideally the release notes are distributed at the same time as the update or shortly after. Having the release notes accessible can help users track updates over time, ensuring transparency [4]. This can be done by for example archiving the past release notes.

2.1.3 Best Practices

Through a review of various online blogs and articles on effective release note practices, I have compiled a list of key elements that can help produce effective release notes. The following sources were used to compile these best practices: [5], [17], [15], [18], [26], [4]. This list aims to provide a comprehensive guideline for creating release notes that are not only informative but also engaging and user-friendly.

- **Use plain language:** Writing in clear, straightforward language can ensure that all users can easily understand it, regardless of their technical expertise. Avoiding jargon can ensure that the information is comprehensible.
- **Keep it short:** Keeping the release notes concise and to the point, highlighting only the most important changes and updates. This helps in maintaining the reader's attention and ensuring that the messages are conveyed.

- **Organize the content:** Using headers, bullet points, and a content index can allow users to quickly navigate to sections of interest.
- **Focus on value for the reader:** Emphasize how the updates and changes benefit the user. Highlight new features, improvements, and bug fixes in a way that clearly show their value and impact on the user experience.
- **Visualize content:** Incorporating visuals such as screenshots, diagrams, or videos can make the release notes more appealing and easier to understand.
- **Include relevant additional information:** Providing links to detailed documentation, tutorials, or related resources can be very helpful. This allows users who need more in-depth information to easily access it without making the main body of the release notes too long.

2.2 Salesbuildr

Salesbuildr is a company that specializes in sales automation software for small to medium-sized enterprises [19]. Their software is used to help businesses streamline their sales operations and enhance productivity. This is done by integrating various tools and existing systems into a cohesive platform, making it possible for businesses to optimize their sales processes efficiently. This platform enables users to manage leads, track customer interactions, and automate routine tasks efficiently. Salesbuildr is an expanding company with 115 companies using its platform, serving over 2000 users in total.

The market of Software-as-a-Service (SaaS), in which Salesbuildr operates, is constantly evolving due to the changing needs of businesses and the competitive landscape. To remain relevant in this fast-paced industry, companies need to keep improving their product, which leads to frequent software updates. These updates aim to enhance functionalities, improve the user experience, and address any flaws in the software. It is therefore imperative for Salesbuildr to effectively communicate these updates through well-written release notes. When these release notes are written clearly and comprehensively, this can help users understand the changes made, which can lead to a higher adoption rate and reduced support tickets.

The purpose of this study is to explore and develop methods to improve the clarity and effectiveness of Salesbuildr's release notes. This will be done by examining the current practices and identifying areas for improvement. The goal for Salesbuildr is to ensure that users are well-informed and can fully benefit from the latest software updates.

2.2.1 Current Situation at Salesbuildr

To gain insights into the current situation of the release notes at Salesbuildr, an exploratory interview was held with the Customer Success Manager. This interview provided valuable insights into how the release notes are currently being communicated and the content they include.

The release notes are disseminated to users through both an in-app notification and an email. The in-app notification directs users to a Salesbuildr website [?], where the bi-weekly release notes are published. Additionally, the email contains the release notes directly within its content. At the top of the web page, an introduction is provided, including a week number and a brief summary of

the release. At the side, a table of content is provided to have a good overview of the discussed items.

In the release note each feature is introduced with a heading, followed by a detailed description. The benefits and relevance to users are addressed through bullet points that outline the advantages of each feature. For some new features, instructions on how to enable or use them are provided. This includes visual aids such as a screenshot, a short clip or a recorded video walk-throughs of new features. Links for further assistance or more in-depth explanation can also be included. After the new features announcements, items Salesbuildr is currently working on are listed. These items are sorted into categories and have a short description on what the improvement or new features will be. There is also an option to review and vote for new feature requests. A new feature request can be sent to Salesbuildr for in the future or requests made by other users can be 'upvoted'.

Initially, the release notes were manually written by the CEO, who was aided by ChatGPT to expand on the items summarized from the release sprints. The information used to write the release notes is sourced from Jira, a project management tool where Salesbuildr's development team organizes and tracks the progress of their software development tasks. The data in Jira includes various fields such as issue type, issue ID, summary, assignee ID, reporter, priority, status, created date, updated date, due date, and sprint. The current time spent on producing these release notes is about one to two hours. The current reach of the release notes is in the months March and April is around 125 views.

When comparing these release notes to the list of best practices for effective release notes, several insights can be gathered. Firstly, Salesbuildr effectively uses plain language, ensuring that the content is accessible to users with varying levels of technical expertise. The document is also relatively concise, focusing on the most important changes and updates, which helps maintain user attention. Additionally, the content is well-organized, featuring headings, bullet points, and a table of contents that allows users to quickly navigate to sections of interest.

Salesbuildr emphasizes the value of updates by outlining the benefits and relevance of each feature through bullet points, which aligns with the guideline to focus on value for the reader. Furthermore, the inclusion of visual aids such as screenshots, short clips, and video walk-throughs enhances the clarity and appeal of the release notes. Finally, the document includes links for further assistance, providing users with additional information. Overall, Salesbuildr's release notes align well with the established best practices.

3 Related Work

Release notes serve as essential documents in the software development process by providing a summarized version of the new developments in the latest release about updates, enhancements, and bug fixes to end-users[11]. Despite their importance, there is a gap in the literature on best practices for crafting effective and engaging release notes. This section reviews relevant research on the creation, communication, and automation of release notes, along with research on user-developer communication, to provide a strong foundation for this thesis.

According to Bi et al. [24] release notes are one of the most important software artifacts in the communication between software development teams and end-users. This study aims to fill a gap in the underexplored area of release notes in software development by investigating the production and use of release notes. Through analyzing a large dataset of release notes from 1000

GitHub projects and conducting interviews and surveys with experts to gather more insight, it revealed discrepancies in perceptions between the producers and users. The study emphasized the importance of well formed release notes and provided practical guidelines for both the creations and utilization of release notes, giving valuable recommendations for enhancing developer-user communication in the software development process.

Agile development methodologies promote frequent releases. However, the rapid cadence can overwhelm users seeking clarity on version disparities. This leads to questions about the necessity of immediate updates, the inclusion of new features or fixes, and the value of trying out a new version. While release notes can address these concerns by providing important information on each release's content, crafting high-quality release notes requires significant time and effort. Klepper [20] states that release managers must consider the target audience, gaining information from project management, issue trackers, build systems and consulting developers and designers. A semi-automated approach is introduced aimed at easing this burden. Their method empowers release managers to curate auto-generated content, leveraging insights from both build servers and issue trackers. Moreover, it facilitates customization of release notes to provide the specific informational needs of various audience segments.

Abebe et al. [22] conducted research on the content and structure of release notes, aiming to address the lack of insights into this underexplored area in research literature. To fill this gap, 85 release notes were manually analyzed across 15 software systems, identifying six types of information types in release notes. This was the first step towards offering more specific guideline for the release notes writers. In addition, another finding was that the writers do not follow a consistent pattern writing the release notes, leading to varying information types listed within. A machine learning approach to automatically identify significant addressed issues that should be included in the release notes in order to help the authors.

Aghajani et al. [6] looked more into documentation for the entire software project instead of only the release notes. Good documentation is valuable as it facilitates stakeholder usage, comprehension, maintenance, and evolution of the system. In practice this has yet to be investigated because it often lacks sufficient content and clarity. Aghajani's research was the first step into developing automated documentation generation recommenders to automatically recommend precise documentation suited for particular tasks. Through conducting two surveys, they aim to investigate practitioners' perceptions of documentation issues and solutions, along with the types of documentation considered important for various tasks.

Moreno et al. [11] conducted a study on automating the generation of release notes in software development. It introduces ARENA (Automatic RElease Notes generAtor) which is designed to streamline the creation of release notes by extracting and summarizing the changes in code and incorporating information from versioning systems and issue trackers. Through empirical studies with 53 participants, the research demonstrates that the ARENA-produced release notes closely align with the manually produced release notes and often include additional important information that may be absent in the manually produced release notes. ARENA presents a promising step forward in automating release note generation, providing a dependable and effective solution that improves the thoroughness and precision of release documentation in software development endeavors.

Ali et al. [12] also did a study on automatic release notes generation. The goal is to see if the automatic produced release notes are as good as the manually created ones, which are more complex and time-consumable. The approach used was in python and generated notes for node.js

projects. They made a four step approach: extract changes, summarize, extract issues and generate doc. The evaluation of the results was done manually and the results showed that the automated release notes were more precise than the ones produced manually.

Another topic important to this research is the effective communication between user and developer. The involvement and participation of users for developing a successful software system is studied to be fundamental. Abelein [1] performed multiple studies on this. Firstly, literature was studied on the User Developer Communication (UDC) and they concluded that enhanced UDC had a beneficial effect on the success of a system. In contrast to existing research, which primarily focuses on user involvement either early or late in the development phases, this study directed attention to the critical stage of translating user requirements by developers. The interpretation by developers and the transformation of users' requirements into technical characteristics is pivotal in the process of software development. During this stage of the process, communication with end users is essential for certain decisions. A method was created that involved identifying modifications to the requirements and determining the optimal level of detail for engaging with end users.

In this thesis, a combination of academic literature and grey literature was used to gain information on the use of release notes. Grey literature are sources that are not formally published in academic journals, such as blog posts, survey results, and technical reports. These sources can provide practical insights and real-world experiences. Garousi et al. [25] say that although researchers primarily trust the academic sources, software engineering practitioners can often rely on grey literature because of the accessibility and practicality. Practitioners can also contribute to grey literature, sharing their findings and experiences and insights. Although grey literature is not new in software engineering industry, the use and benefits in empirical software engineering research are increasingly recognized.

4 Method

This section outlines the research methodology employed in this thesis, which seeks to address the research question: "How can software release notes be made more effective in reaching and engaging users?" To investigate this, several sub-questions were formulated:

- What are release notes?
- How are the release notes currently being communicated at Salesbuildr?
- What is the current reach of Salesbuildr's release notes?
- What are the generally accepted best practices for creating effective release notes?
- How do Salesbuildr's current release notes compare with these best practices?
- What specific improvements can be made to Salesbuildr's release notes to enhance their effectiveness?
- How can these improvements be implemented in Salesbuildr's release notes process?

Most of these sub-questions have already been addressed in the background section, providing necessary context to inform the research approach. The answers provided a foundation to help determine the areas for improvement in the release notes to address the research question. To answer the research question, a mixed-methods approach was employed, grounded in the design science research (DSR) paradigm. This methodology focuses on the creation and evaluation of artifacts designed to solve identified problems, making it suitable for developing and evaluating the proposed solutions to enhance release notes.

4.1 Design Science

To conduct the research the paradigm of design science research (DRS) methodology was used. Design science research focuses on the construction and evaluation of an artifact, designed to solve the at the start determined problem. This approach has been acknowledged as an important and legitimate research paradigm within the Information Systems research domain[7]. Unlike traditional research methodologies that often focus on the understanding and explanation of instances, DSR is characterized by its focus on the construction and evaluation of artifacts, models, and methods[8]. It focuses on generating new insights and contributions to knowledge by examining the designed artifacts and the process used to create them[8].

When looking at DSR framework in Figure 1 made by Peffers et al. [10] the research method consist of six steps, of which the first one is problem identification and motivation. Here the problem is defined and its importance is explained. The next step is to define the objective for the solution. What objectives must the artifact meet to solve the problem effectively? When you have a clear objective the design and development stage begins. The architecture and functionalities need to be determined here as well to be able to create the artifact. After this, a demonstration shows how the designed artifact solves the problem. This is often done through simulations, experiments or case studies. On the evaluation, the artifact is assessed on its effectiveness and efficiency. Based on feedback and results the artifact can be adjusted as needed. The final step in this process is the communication, where documenting and representing the research process and its findings to both academic and practitioner audiences. In the figure below the DSR can be seen as an iterative process, as there is a loop from evaluation and communication back to the definition of objectives and the design steps.

In the Figure 1, you can see the different entry points into the DRS process. These entry point indicate that a design science research project can begin at various stages, depending on the researchers starting point and the context. The four entry point are as follows: 'problem-centered initiation', where research starts with identifying a specific problem; 'objective-centered solution', this begins with defining the objectives for a solution; 'design and development centered initiation', which starts directly with creating and developing an artifact; and 'client/context initiated', where the process starts with an existing artifact and focuses on evaluation and improvement. These entrance points demonstrate that DSR methodology is flexible and researchers can use this methodology at various stages based on their research context.

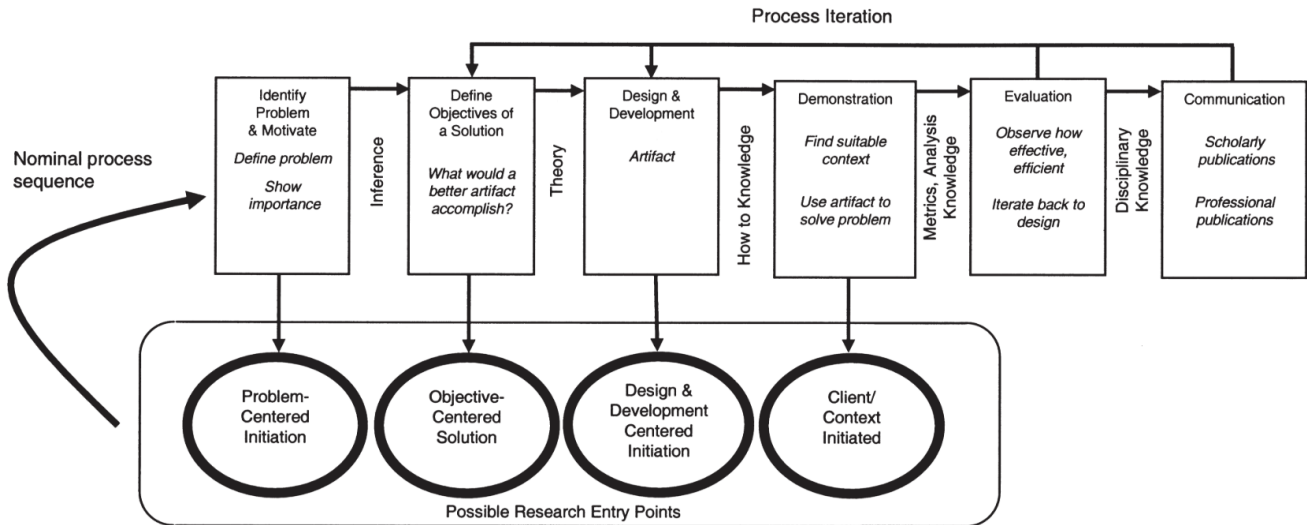


Figure 1: Design Science Research Methodology (DSRM) Process Model by Peffers et al. [10]

4.2 Methodological Approach

For this thesis, the design science research methodology was chosen because it provides a structured framework for solving real-world problems through the development of tangible artifacts. This approach aligns with the thesis objective to evaluate the effectiveness of the current communication of the release notes and to explore new methods for producing and communicating these release notes. My approach for this is illustrated in figure 2

The entry point for this research was problem-centered initiation, where the incentive for this problem was specified. After determining the problem, more information was gathered to gain better insights into the current processes for communicating the release notes. To achieve this, a survey was conducted along with a few follow-up interviews with users of Salesbuildr. This helped in defining the research objectives and establishing some requirements for a possible solution.

To examine the current state of the release notes at Salesbuildr a mixed methods approach was used. This consisted of a survey and follow-up interviews. The survey aimed to understand user satisfaction and gather feedback on the content and structure of the bi-weekly published release notes. To build the survey, Qualtrics, a survey tool used at Leiden University, was used. It included questions that gave some guidance on the current satisfaction levels and the main reasons the users read the release notes. This helped identify the current bottlenecks and provided new insights from a user perspective.

After gathering these insights, an artifact was designed as a potential solution to the problem. A virtual walk-through was developed to explain new features comprehensively. This artifact was then evaluated by both the company and the platform’s users. It was implemented in the release notes to assess whether its inclusion could enhance the reach and effectiveness of these notes. This research not only involves designing and demonstrating the artifact but also integrating it into the release notes for practical evaluation.

The evaluation of the artifact’s effectiveness was measured in two ways. First, by assessing whether the reach of the release notes and the virtual walk-through had improved. Secondly, textual feedback was collected from customers who participated in follow-up interview after the survey.

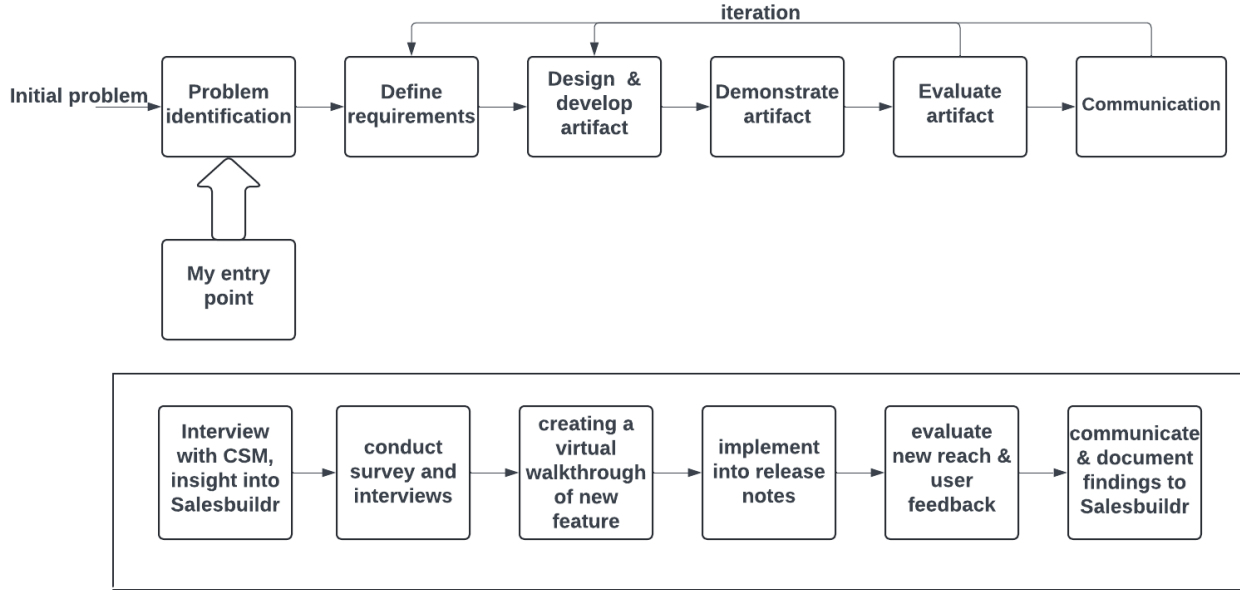


Figure 2: This study’s design science research approach

These combined insights provided an understanding of the effectiveness of the developed artifact. From this, conclusions were drawn for the company on the continued use of this method in the communication of new features.

5 Definition of Requirements

In this chapter, the outcomes and results of user feedback and interaction data are discussed, forming the foundation for the requirements used in developing the artifact. By analyzing survey responses and interview insights, areas for improvement are identified in the content, format, and overall user experience of the release notes.

5.1 Survey

A survey was sent out to all admin-users in the Salesbuildr application, approximately 411 users, from whom 36 responses were received within two weeks, resulting in a response rate of 8.7%. While this might seem low, considering the 115 companies using the platform, the response rate per company is around 30%, which is significantly higher than anticipated.

The survey results revealed several key insights into how users interact with and perceive the release notes. Approximately 70% of respondents work in the sales department, highlighting the primary user demographic that engaged with the survey. This insight can be seen as significant as it can indicate that the provided information must be relevant and beneficial for the sales professionals who use software features to enhance their productivity. Furthermore, 58% of the respondents answered that they read the new release notes every two weeks. This can be interpreted as a high level of engagement and interest of the respondents in staying informed on the updates. In contrast,

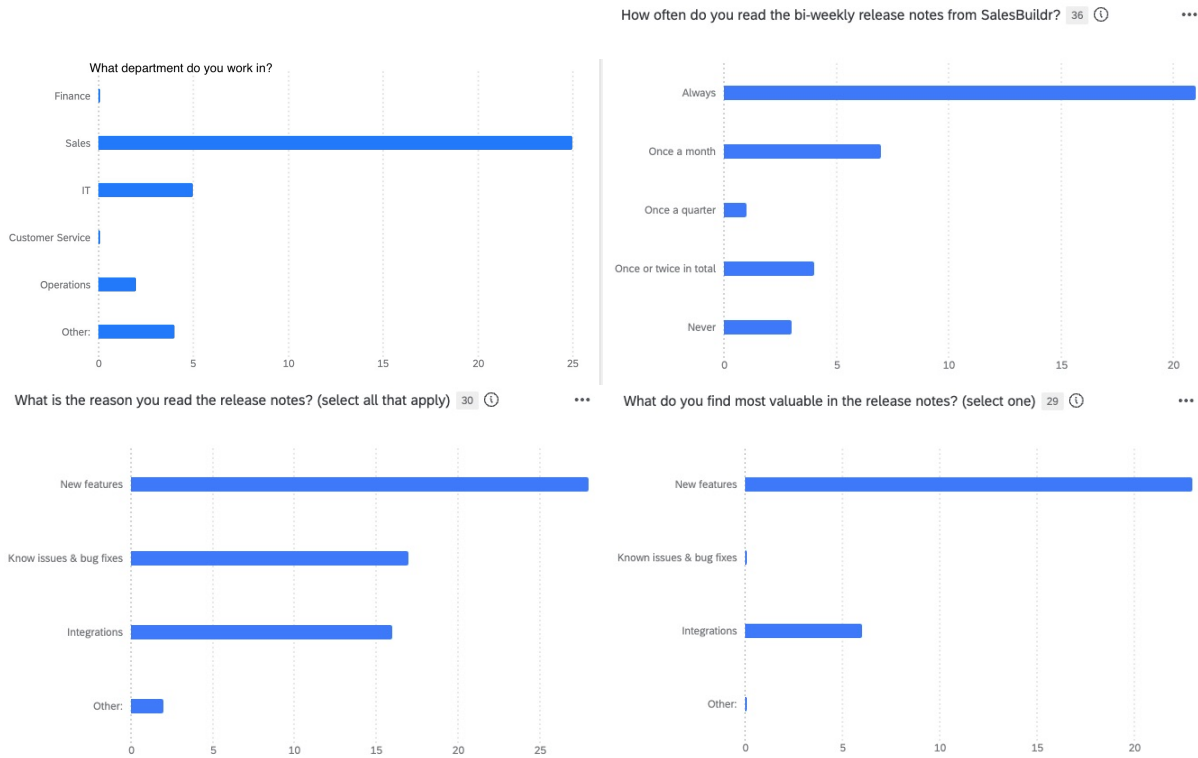


Figure 3: visualisation of the gathered data from the survey

only 8% of the respondents have never read any release notes, this can suggest that the majority of users have some sort of value in the release notes.

The content within the release notes that respondents find most valuable includes new features and integrations. The reason for this might be that understanding the latest functionalities can be used to improve performance. When asked to prioritize, nearly 80% of users chose 'new features'. This can indicate a strong preference for understanding new capabilities over new integrations. This information can be important for tailoring future release notes to focus on what users find most beneficial. Visualisations of the gathered data from the section above can be seen in Figure 3

Respondents were asked to rate their satisfaction of the content in the release notes on a scale from 1 to 5, where 1 indicates very dissatisfied and 5 indicates very satisfied. The results showed an average satisfaction score of 4.29, with the lowest score being 3. The high average score can suggest that overall the respondents are pleased with the content in the release notes, indicating that the information is relevant and meets their needs. However, the fact that some respondents rated their satisfaction as 3 in the first graphic in Figure 4 suggests there may be room for improvement to fully meet the expectations of all users.

In terms of findability of information, respondents were also asked to rate this aspect on a scale from 1 to 5. As seen in the second graphic in Figure 4 the average score was 3.97, with the lowest score being 2. While this score can indicate that most respondents find the information relatively easy to locate, the lower scores point to occasional difficulties that some respondents may experience when navigating the release notes. This can suggest a need for improving clarity and possibly a better organization of the content to ensure all users can find the information effortlessly.

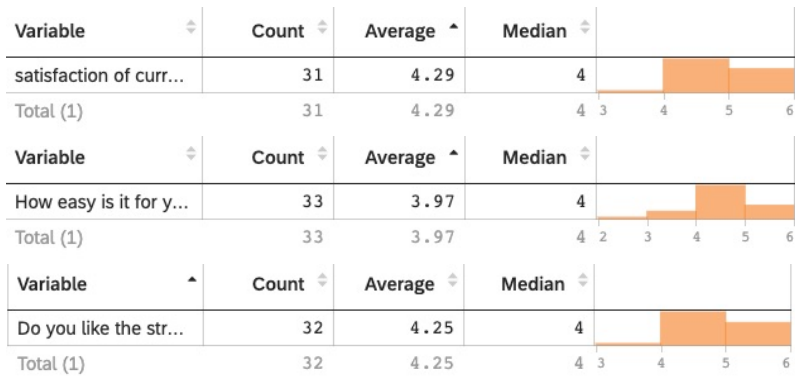


Figure 4: visualisation of the rating questions of the survey

The structure of the release notes received an average score of 4.25 from the respondents, indicating that the layout of the release notes are generally well-received. A well-structured release note can help users quickly understand and assimilate the new information. This high score reflects positively on the current structure of the release notes used by Salesbuildr, but the variance in ratings also hints that there are potential areas for improvement to better serve all users. This result is visualised in the last graphic of Figure 4.

5.2 Interviews

At the end of the survey, respondents were invited to express their interest in participating in follow-up interviews. The objective of these interviews was to gain a better understanding of their experiences and perceptions regarding the release notes. Sixteen respondents indicated their willingness to participate in these follow-up sessions. Subsequently, these respondents were contacted through email, and four of them were available in the next two weeks. Interviews were conducted via the Google Meet platform to enable remote discussions. Before the interviews, an email was sent to the selected users, detailing eight topics to prepare them for the conversation. The topics included:

- Improvement in Release Notes Content
- Ineffective Communication Channels
- Challenges in Locating Information within Release Notes
- Ideal Structure for Release Notes
- Handling Inconsistencies or Discrepancies
- Missing Features or Functionalities in Release Notes
- Effective Release Notes Examples
- Future Role of Release Notes

The interviews, each lasting approximately twenty minutes, provided valuable qualitative data that augmented the results of the survey. Findings from the interviews included high user satisfaction with Salesbuildr compared to other platforms, adequate communication channels, and an appropriate frequency of release notes to keep users engaged. Users appreciated the visual aids and structured content but found the volume of information sometimes overwhelming.

From these conversations, several suggestions for improving the release notes and Salesbuildr’s platform were made. These suggestions ranged from small content changes to the introduction of new features. One significant insight gathered from the interviews indicated the potential benefit of implementing a virtual walk-through instead of a recorded video to introduce new features. This proposed solution aims to make the release notes more engaging and informative, with the goal of improving user satisfaction and engagement.

All findings, including aspects not directly related to the release notes, have been discussed with the Salesbuildr team to determine feasible improvements for future implementation. These insights cover various aspects of the release notes and platform functionality, aiming to enhance overall user experience. The quantitative data gathered from the survey combined with the qualitative insights from the interviews provides a good understanding of the users’ needs and preferences.

6 Design and Development

The purpose of the artifact is to enhance communication of the release notes by implementing a virtual walk-through feature. Insights from surveys and interviews suggested for a more intuitive approach to introduce and navigate the updates. This section outlines the conceptualization, design, development, and integration of the virtual walk-through.

The primary goal of the artifact was to offer users with an interactive and intuitive tool to explore new features at their own pace. Drawing inspiration from interactive tutorials found in other software platforms, an analysis of requirements preceded the design phase. Functional requirement included enabling users to navigate features step-by-step, incorporating media components such as screenshots or videos, and incorporating interactive elements. Non-functional considerations are compatibility across various devices, scalability and responsiveness. Additional factors included ensuring visual appeal, ease of use, and directly accessible within the release notes.

For the creation of the artifact, Storylane was used. Storylane specializes in enabling businesses to create and share interactive product demos. As a no-code platform, it offers the advantage of allowing rapid and straightforward creation of demos. According to their documentation [21], demos can be built in ten minutes, making it an efficient choice for developing the interactive walk-through. This tool was chosen because it aligns well with the project’s need for a user-friendly interface that supports quick iteration and deployment of interactive features. Its capability to integrate multimedia components such as screenshots was beneficial for the enhancement of user engagement and to understand new features.

To utilize Storylane effectively, the first step involves installing their browser extension. Activation of this extension starts the recording process within Storylane, capturing screenshots with each click during the walk-through of the new feature. Upon completion, clicking the extension again finalizes the recording, which then becomes accessible within the Storylane application or browser interface. Here, adjustments could be made to the sequence, annotations added, hyperlinks embedded, styles customized button configurations set, audio overlays recorded, and navigation

controls integrated. Users could progress through the demo by either clicking on recorded points or utilizing provided navigation buttons.

After completing the demo, it was integrated into the Salesbuildr week 22 release notes by embedding a hyperlink. This allowed users to access the virtual walk-through directly from the document. Ideally, the walk-through would have been embedded directly into the release notes without requiring a hyperlink, but due to time constraints, this was not feasible at the moment. Implementation enabled users to interact with and explore the new feature demonstration within the release notes, potentially enhancing their understanding and overall user experience.

7 Evaluation

Two approaches were used to evaluate the results of the virtual walk-through implementation. The first involves using analytics from Storylane and Intercom to assess the time spent and engagement metrics to determine the effectiveness of adopting the artifact. The second approach includes analyzing the textual feedback obtained from interviewees. Combining these two methods can provide a comprehensive assessment of the effectiveness of the virtual walk-through.

7.1 Quantitative Evaluation

To gather data on the interaction with the walk-through multiple methods were employed. Storylane's built-in analytics feature provided insights such as company identity, last access, completion status, time spent, and engagement with the call-to-action (CTA). The results indicated a total of seven views from distinct companies within the period spanning from the release notes publication to two weeks thereafter. The duration of engagement with the walk-through varied significantly, with most users spending between one to two and a half minutes, while one user invested approximately eight minutes, and another needed one hour and 45 minutes to complete the walk-through, which makes it an outlier and can indicate they were doing something else. Completion rates were high, with six out of seven users achieving 100%, while one user achieved an 84% completion rate. Additionally, three out of the seven users interacted with the CTA during their session.

Furthermore, assessing the reach and impact of the release notes was crucial to understand the dissemination among users. Insights into the reach of the release notes were gathered through Salesbuildr's analytics in Intercom, their customer engagement platform. These analytics provided insightful data on the number of views and interactions with the release notes. Notably, the release notes during week 22 stood out as an outlier, with approximately 240 views recorded.

To evaluate the efficiency of implementing the virtual walk-through, you have to consider the time invested in creating the artifact. Currently, crafting the release notes requires approximately one hour. The original method to explain features in the release notes involved creating video recordings, which typically takes between two to five minutes. In contrast, developing the walk-through spans from ten to fifteen minutes. These estimates exclude the additional time spent familiarizing oneself with the Storylane platform, which took approximately twenty minutes.

7.2 Qualitative Evaluation

For the second part of the feedback, qualitative data was provided by users who also participated in the interviews. This feedback was collected through email after notifying the interviewees when the release notes were published. Users provided feedback on the intuitiveness, user experience, potential improvements, and recommendations for broader use.

The feedback received on the virtual walk-through demo was positive and highlighted the effectiveness in enhancing user experience and learning. Users appreciated the transition from traditional video explanations to the interactive Storylane demo structure, emphasizing the clarity, sharp visuals, and ease of comparison with their own Salesbuildr interface. One user described it as a "game changer" due to its good integration with their own workflow, eliminating the need for frequent pauses of the recorded video.

Moreover, the flow of the walk-through received positive feedback for its intuitiveness, providing a straightforward learning experience with clear prompts that eased the navigation. Although one user noted a slight initial confusion in navigation, attributing it to personal familiarity, overall, the walk-through was deemed intuitive and user-friendly. Another user expressed interest in expanding the use of Storylane demos across different fields, suggesting the potential for a broader application of the use of it within the platform.

In addition to these usability insights, the feedback indicated no specific suggestions for further improvement to the demo, reflecting satisfaction with the current functionality and structure. This positive reception underlines the potential of Storylane as a valuable tool for creating interactive learning materials, with suggestions for future implementations in for example user manuals.

8 Results

A conclusion can be drawn by analyzing the time spent on creating the release notes, including both recorded videos and virtual walk-through, and the reach of those release notes. Additionally, user feedback is considered to help determine the effectiveness of the artifact in enhancing the user experience. By examining these metrics, the evaluation aims to provide a clear assessment of the efficiency and effectiveness of the virtual walk-through.

| | Time spent on feature introduction (minutes) | Time for composing the release note (minutes) | Total time spent (minutes) | Reach |
|------------------|--|---|----------------------------|-------|
| Without Artifact | 5 | 85 | 90 | 130 |
| Using Artifact | 15 | 60 | 75 | 243 |

Table 1: Comparison of time spent and reach with and without the artifact

Table 1 presents a comparative analysis of the time spent on introducing a single feature and the overall time invested in composing the release notes, alongside the reach achieved by each method. This data helps the evaluation of the effectiveness of the artifact. The original method, involving a video recording, requires approximately 5 minutes for the feature introduction and an additional 85 minutes for composing the release notes, resulting in a reach of 130 users. These measurements are approximations and were made at the onset of the research. In contrast, the

virtual walk-through method needed 15 minutes for the creation and the release notes are composed in 60 minutes, making the total 75 minutes. This method had a reach of 243 users.

To evaluate the effectiveness of the artifact, the following factors are considered:

1. Time Efficiency:

- Without Artifact: Total time spent is 90 minutes.
- With Artifact: Total time spent is 75 minutes.
- Time Saved: $90 - 75 = 15$ minutes.
- Percentage Decrease in Time:

$$\left(\frac{90 - 75}{90}\right) \times 100 \approx 16.67\%$$

2. Reach:

- Without Artifact: Average reach of 130.
- With Artifact: Reach of 243.
- Increase in Reach: $243 - 130 = 113$ users.
- Percentage Increase in Reach:

$$\left(\frac{243 - 130}{130}\right) \times 100 \approx 86.92\%$$

3. Reach per minute:

- Without Artifact: $\frac{130}{90} = 1.44$ users per minute.
- With Artifact: $\frac{243}{75} = 3.24$ users per minute.
- Increase in Reach per Minute: $3.24 - 1.44 = 1.8$ users per minute.
- Percentage Increase in Efficiency: $\left(\frac{3.24 - 1.44}{1.44}\right) \times 100 \approx 125\%$.

To summarize, the evaluation highlights the artifact's effectiveness in improving efficiency and user engagement. Despite requiring slightly more creation time, the virtual walk-through reduces overall release note crafting time by 15 minutes. Additionally, it expands the reach of the release notes by 113 users, illustrating the ability to increase the reach with 87%. The artifact improves reach per minute spent on making the artifact from 1.44 users without it to 3.24 users with it, marking a significant 125% increase in efficiency in communicating new features.

Overall, these findings accentuate that adopting the virtual walk-through method is a more efficient and effective approach for introducing new features in the release notes. The positive impact on reach and user satisfaction suggests that further development and integration of interactive demos could provide further advantages for improving user engagement and overall experience within Salesbuildr.

9 Discussion

This section explores several factors that may have influenced the implementation and outcomes of the artifact, examining both its effectiveness and limitations. These discussions are organized chronologically based on the research conducted.

Initially, the study encountered a limitation in survey participants, with only 36 respondents. This sample size is insufficient to draw statistically significant conclusions, highlighting the need for future research on a larger scale to ensure accurate findings.

After the research began and measurements were taken, Salesbuildr appointed a Communication Channel Chief. This role aims to enhance user and stakeholder awareness of Salesbuildr updates and features. This appointment could have contributed to the observed increase in reach during the artifact's measurement phase.

Another critical consideration is the deviation from the intended implementation of the artifact within the release notes. Originally conceived as a visual feature integrated directly into the release notes interface, the actual implementation utilized a hyperlink to access the virtual walk-through. This deviation may have resulted in reduced views and engagement, potentially leading users to overlook the virtual walk-through. Consequently, the recorded views of the demo on Storylane amounted to only seven, suggesting that the true reach and impact of the artifact could be underestimated due to this implementation method.

To evaluate the time required for creating the release notes with the artifact, input was sought from the CEO, who initially wrote the release notes for this research. The estimation was that currently it takes around 60 minutes to craft the release notes. Following the appointment of the COO, who now assists in the process, this can be the reason the time has been reduced. Previously, during the initial measurements, the creation time ranged from one to two hours, prompting the evaluation to use 90 minutes as a benchmark. This reduction in time due to collaborative efforts may impact the accuracy of the evaluation's conclusions.

The conclusion drawn from this research suggests that the artifact effectively enhances user engagement. However, these findings could have been influenced by several other factors or significant changes, such as the introduction of a long-awaited feature or a surge in new users within Salesbuildr in recent months. The substantial increase in reach may have been driven by these external factors rather than solely by the artifact itself. Additionally, Storylane reported only seven views, indicating that a small number of users engaged with the demo. This minimal visibility suggests that the artifact's impact on the overall reach may be limited, questioning the direct influence on the observed increase.

Finally, the findings of this study might not be universally applicable to all companies. The sample size of 36 survey respondents is relatively small, which limits the generalizability of the results. Additionally, the context of Salesbuildr, including its user base and the specific feature being introduced, may not reflect the circumstances of other companies. Different organizations may have varying levels of user engagement, different communication strategies, and other product features, all of which could influence the effectiveness of similar artifacts. Therefore, while the results provide valuable insights for Salesbuildr, further research with a larger and more diverse sample is necessary to determine whether the observed benefits can be replicated in other settings.

10 Conclusions and Future Research

This research aimed to address the question: "How can software release notes be made more effective in reaching and engaging users?" Through this study, it was found that while users were generally content with the release notes produced by Salesbuildr, there was still room for improvement. The solution focused on substituting the recorded videos of new features with a virtual walk-through. This virtual walk-through was designed, developed, and implemented into the release notes.

The artifact was then evaluated both qualitatively and quantitatively. The quantitative results showed an 87% increase in reach; however, the views of the artifact did not indicate a definitive cause for this increase. Qualitative user feedback was very positive, strongly suggesting that the virtual walk-through was effective in enhancing communication. Therefore, the research concludes that implementing a virtual walk-through can increase the effectiveness of release notes. However, these results must be interpreted with caution due to several discussed limitations, which raise some concerns on the findings.

Future research could address these limitations and further explore the impact of virtual walk-throughs in different environments and with more variation in user groups. A larger, more diverse sample could help make the findings more generalizable across different contexts. Furthermore, implementing the artifact as a visual in the release notes, as originally intended, and surveying its impact over a longer period could provide more accurate insights into its effectiveness. Understanding the specific user behaviors and preferences that contribute to engagement with virtual walk-throughs would be crucial for optimizing their design and integration into software update communications.

For Salesbuildr, future research could investigate the factors that contributed to the significant increase in reach. Analyzing these factors would offer insights into further improving user engagement and adoption rates of software updates.

Finally, exploring other methods for presenting release notes, such as interactive tutorials or gamified explanations, could provide additional insights into enhancing user engagement and satisfaction. These alternative methods could help discover different learning styles and preferences among users, which could potentially increase the effectiveness and accessibility of release note communication.

References

- [1] Ulrike Abelein and Barbara Paech. State of practice of user-developer communication in large-scale it projects. In Camille Salinesi and Inge van de Weerd, editors, *Requirements Engineering: Foundation for Software Quality*, pages 95–111, Cham, 2014. Springer International Publishing.
- [2] Antrika. What is changelog? best practices to write effective release notes. <https://medium.com/@antrika.gm/what-is-changelog-best-practices-to-write-effective-release-notes-83870036e86b>, 12/02/2024. Accessed: 23/05/2024.
- [3] Parag Bhole. Software release notes : Definition, overview and importance. <https://amoeboids.com/blog/release-notes-complete-guide/>, 27/05/2024. Accessed: 14/06/2024.
- [4] Jake Brereton. Who writes release notes: A comprehensive guide. <https://www.launchnotes.com/blog/who-writes-release-notes-a-comprehensive-guide>, 21/06/2023. Accessed: 23/05/2024.
- [5] Clickup. How to write release notes (with templates examples). <https://clickup.com/blog/how-to-write-release-notes/>, Jan 2024. Accessed: 14/06/2024.
- [6] Mario Linares-Vásquez Laura Moreno Gabriele Bavota Michele Lanza David C. Shepherd Emad Aghajani, Csaba Nagy. Software documentation: The practitioners’ perspective. In *International Conference on Software Engineering*, 2020.
- [7] S. Gregor and A. R. Hevner. Positioning and presenting design science research for maximum impact. In *MIS Quarterly*, pages 337–355, 2013.
- [8] Robert Wayne Gregory. *Design Science Research and the Grounded Theory Method: Characteristics, Differences, and Complementary Uses*, pages 111–127. Physica-Verlag HD, 2011.
- [9] Ayush Jangra. How to write release notes: Best practices, examples and tools. <https://supahub.com/blog/write-release-notes#Who-writes-Release-Notes->, 13/04/2024. Accessed: 23/05/2024.
- [10] Marcus A. Rothenberger Ken Peppers, Tuure Tuunanen and Samir Chatterjee. A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77, 2007.
- [11] M. D. Penta R. Oliveto Andrian Marcus G. Canfora Laura Moreno, G. Bavota. Automatic generation of release notes. In *SIGSOFT FSE*, 2014.
- [12] Wasi Haider Buttt Mubashir Ali, Asad Aftab. Automatic release notes generation. In *IEEE 11th International Conference on Software Engineering and Service Science (ICSESS)*, 2020.
- [13] Gail C. Murphy. Beyond integrated development environments: adding context to software development. In *Proceedings of the 41st International Conference on Software Engineering: New Ideas and Emerging Results*, ICSE-NIER ’19, page 73–76. IEEE Press, 2019.

- [14] Adams B. Khomh F. Engström E. Petersen K. Mäntylä, M. V. On rapid releases and software testing: A case study and a semi-systematic literature review. In *Empirical Software Engineering*, volume 20(5), page 1384–1425, 2014.
- [15] Anand Patel. How to write release notes (template +5 great examples). <https://www.appcues.com/blog/release-notes-examples>. Accessed: 13/06/2024.
- [16] Priyanka. How to write release notes: 10 best release notes examples+template. <https://www.productlogz.com/blog/how-to-write-release-notes>, 10/10/2023. Accessed: 23/05/2024.
- [17] ProductPlan. Release notes. <https://www.productplan.com/glossary/release-notes/>, Sep 2020. Accessed: 13/06/2024.
- [18] Rapidr. How to write release notes: Top tips, examples tools. <https://rapidr.io/blog/write-release-notes/>, Apr 2024. Accessed: 14/06/2024.
- [19] Salesbuildr. <https://www.salesbuildr.com/>, 22/05/2024. Accessed: 22/06/2024.
- [20] B. Brügg Sebastian Klepper, Stephan Krusche. Semi-automatic generation of audience-specific release notes. In *2016 IEEE/ACM International Workshop on Continuous Software Evolution and Delivery (CSED)*, 2016.
- [21] Storylane. <https://www.storylane.io/>. Accessed: 22/06/2024.
- [22] Nasir Ali Ahmed E. Hassan Surafel Lemma Abebe. An empirical study of software release notes. In *Empir Software Eng 21*, page 1107–1142, 2016.
- [23] Blake Thorne. 51 of the best release notes examples (+ 11 free templates). <https://www.launchnotes.com/blog/release-notes-examples>, 2021. Accessed: (April 14 2024).
- [24] David Lo J. Grundy Thomas Zimmermann Tingting Bi, Xin Xia. An empirical study of release note production and usage in practice. In *IEEE Transactions on Software Engineering*, 2022.
- [25] Mika V. Mäntylä Vahid Garousi, Michael Felderer and Austen Rainer. *Benefitting from the Grey Literature in Software Engineering Research*, pages 385–413. Springer International Publishing, Cham, 2020.
- [26] Hilal Çökeli. Release notes best practices: 7 great examples and templates. <https://userguiding.com/blog/release-notes>, Dec 2021. Accessed: 13/06/2024.