Cartoon-Assisted Requirements Engineering (CARE): An Artistic Technique Together and Analyze Software Requirements

Mr. Yousuf Hotak[1]
Ms. Fatima Huqooqee[2]

ABSTRACT

Requirements play a critical role in the software engineering process. They serve as the foundation for the design, development, and testing of software systems. However, the complexity of software nowadays makes it difficult for engineers to specify the customer's needs. Furthermore, documenting software requirements can often seem tedious and uninteresting to both engineers and end-users. Since change requests are inevitable during the software development process, this study aims to suggest a feasible approach for the requirements gathering process. It efficaciously assists in creating a responsive software product as well as increases the level of rapid software delivery, which is deemed as one of the most sought-after characteristics in the field of software engineering. The findings of this study demonstrate that integrating cartoons into the requirements gathering process not only enhances the clarity and precision of requirements but also significantly boosts stakeholder engagement and satisfaction.

Keywords: *Software Engineering, Requirements Engineering, Comic Strips, Software Process Models, Graphic Design and Artificial Intelligence.*

1 Introduction

Software requirements engineering helps to reduce the risk of project failure by ensuring that the software being developed is aligned with the needs of its stakeholders and it meets their expectations. It also helps to minimize rework and reduce the cost of development by ensuring that customer's needs are well-defined and clearly understood before the development process begins. Nevertheless, if the requirements gathering process failed to collect the actual needs of the customer, this could lead to a catastrophe. Additionally, there are many different models proposed for the software development process, such as Exploratory Software Development, Waterfall model, V-model, Prototyping Model, Incremental model, XP, Scrum and others. However, each model has its own style of gathering requirements [10]. For instance, in Waterfall models the requirements are gathered upfront and this model gives no chance to change the requirements in upcoming phases, whereas Agile models give this opportunity to bring changes to the requirements in next stages as well [1].

This research aims to explore the use of cartoons as a novel approach to enhance the requirements gathering process. The primary objective is to assess whether the integration of visual aids, specifically in the form of cartoons, can improve the accuracy, clarity, and engagement in the requirements engineering process.

[1,2] *Faculty of Computer Science, Jahan University, Kabul, Afghanistan*
[1] *Yousuf.hotak@yahoo.com,* [2] *Fatimahuqooqee01@gmail.com*

The research questions focus on evaluating the effectiveness of cartoons in reducing ambiguity in requirements and their impact on stakeholder communication and involvement.

1.1 Cartoons and Animation

Cartoons (also known as comic strips) typically feature recognizable characters, expressions, and gestures to convey emotions and messages [13]. Since cartoons are very understandable for human beings in any culture, it can be found in a variety of media, including comic strips and books, animated TV shows and movies, and editorial illustrations [11].

The history of cartoons can be traced back to early human cave paintings and hieroglyphics, but the modern form of the cartoon as a visual form of humor and satire became popular in the late 19th and early 20th centuries with the advent of newspapers and magazines. Today, cartoons continue to be a popular form of entertainment and communication, as well as appealing to audiences of all ages [12].

When people hear information, they are likely to remember only 10 percent of that information three days later. On the other hand, if a related image is shown, people retain 65 percent of that specific information three days later [11]. In addition, using visual aids in the classroom help improve learning by up to 400 percent [8]. Moreover, eye-tracking research revealed that users pay closer attention to photos, images and comic strips than text; there is a famous saying “Humans are visual creatures” [16]. Likewise, Tweets with images and visual aids received 150% more retweets than text [14]. Meanwhile, those who follow directions with illustrations do 323 percent better than those who follow directions without illustrations [15]. Simultaneously, when an image is displayed for only 13 milliseconds before the next image appears, a part of the brain processes the image longer than the amount of time it was seen [6].

2 Problems in Existing Models

In exploratory software development, there is no standard, structured, or formal method for gathering software requirements [1]. This could lead to huge failures in medium-to-large sized projects.

In the Waterfall model, on the other hand, the customer's needs must be known and defined beforehand, which makes managing software updates through stakeholders very challenging. This can inflate development costs and late product delivery. The amount of documentation required in this model is often extensive, making it difficult for both stakeholders and engineers to understand the project requirements [5]. Additionally, the V-model typically requires even more documentation, which can make it even harder for software developers and customers to understand the project requirements. Furthermore, if the customer's needs change, both the development and testing phases will need to be updated and reworked [3].

In the Prototyping model, creating a sample can be very expensive, and the prototypes may become over-engineered, as designers may make a simple prototype overly complex [9]. On the other hand, in the Incremental model, the requirements are divided into multiple parts, built incrementally, and provided to the customer. However, stakeholders may request changes in this specific process, and the requirements in this model may be both clear and ambiguous [1]. Furthermore, in the Identification phase of the Spiral model, which is commonly referred to as requirement gathering phase, the customer's needs may be ambiguous, and the stakeholder's involvement may be minimal [2]. In summary, requirements are often vague and uncertain in all Waterfall models.

Agile models are not immune to weaknesses and drawbacks when it comes to identifying stories (also known as requirements). For example, some deficiencies of XP in collecting requirements are lack of documentation, limited scalability (as this approach is designed for small-to-medium sized projects), and narrow focus (as XP prioritizes software delivery speed, which can lead to a lack of focus on gathering requirements) [7]. On the other hand, in the Scrum model, lack of detailed documentation, dependence on an experienced and skilled product owner and face-to-face communication could pose difficulties in assembling software needs [4]. Similarly, in the Feature-driven development model, complexity, time consumption, and lack of collaboration between stakeholders and developers can lead to inadequate gathering of customer's needs. In conclusion, stories could be confusing and unclear in all Agile methodologies [4].

3 Proposed Method/Developed Technique

As we discussed in the previous section, both engineers and stakeholders often face challenges while collecting customer's needs. Traditional methods of requirements gathering can be tedious, ambiguous, and time-consuming. However, the use of cartoons can help reduce the risk of misunderstandings and miscommunications; as the visual representation of customer's needs can be more easily understood and remembered. Additionally, using animations to gather requirements in software development can make the process more engaging and accessible to stakeholders who may not have a technical background. By incorporating visual aids in the form of comic strips, complex ideas and requirements can be clearly communicated to a wider audience.

During the process of gathering requirements, expert and skilled graphic designers can assist the team in converting the story into animated films as soon as the needs are collected from customers in a series of meetings. In this proposed method, the basic requirements must be collected upfront.

As mentioned, in the software development life cycle, the requirements gathering and analysis phase form the foundation of the software; so, this stage must be taken seriously by software engineers. The CARE process is not merely helpful for requirements gathering, but also for analyzing them simultaneously with customers through animated pictures. However, it is noteworthy that changes are easily managed in the middle of the process since cartoons are editable and can be modified rapidly. This requirements practice is applicable for both waterfall and agile models. Furthermore, after the requirements gathering process is finished, the animated film could be exhibited to the development team, such as designers, programmers and testers. With the help of cartoons, the overall concept of the system is cleared and ambiguity could be

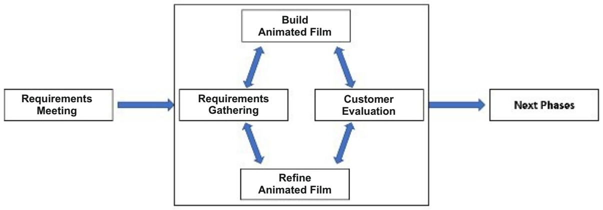


Figure-1: CARE Process Diagram

removed completely. The CARE process diagram is shown in Figure-1.

Requirements Meeting: In this stage of CARE process, the project manager arranges a series of meetings with the investors, customers, stakeholders and end-users alongside the requirements engineers and graphic designers in order to gather the requirements. It is noteworthy that other team members, such as software designers and developers can also attend the meeting in order to gain an idea about the system. The whole CARE process could take 4-6 weeks, depending on the complexity of the software.

Requirements Gathering: The looping stages of the CARE process starts here. In this stage, the customers and end-users describe the needs of the system and the requirements engineers write the requirements documents. Furthermore, The system must be divided into multiple modules, and the most important module must be picked with the consultation of the meeting members in order to be developed. The overall concentration must be on that specific module and the system story will be told by the particular end-user.

Build Animated Film: After the description of the needs by end-users, the graphic designers make animated films accordingly. The number of graphic designers required can be determined by the project manager based on the complexity of the software. Additionally, it is possible for graphic designers to create a simple 2D cartoon in a matter of minutes, especially if they already have pre-made assets such as characters and backgrounds that they can work with. A simple 2D cartoon could consist of a short sequence of drawings that are animated to convey a story or a message.

Customer Evaluation: In this stage, customers and other end-users watch the animated film which is made by the graphic designers and give their feedback in order to alter and modify the story.

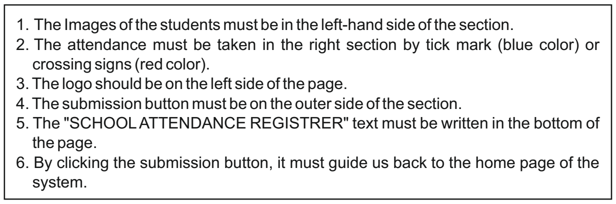
Refine Animated Film: After the evaluation is done by the customers and changes are brought, the film is being refined by the graphic designers, which gives an overall understanding to the members and makes good cooperation among the development team. Additionally, after the first module is turned into an animated film, the second most important module will be picked and passed from the same stages until the whole system is turned into an animated film.

Next Phases: After the animated system is accepted by the customers and end-users, the next phases of development start for that specific system (if waterfall model is used) or specific module (if agile models are used), such as, design, code, test.

To illustrate the CARE concept as a case study, we will discuss the student's attendance in the School Management System.

3.1 School Management System

In this system, the entire student's data is recorded in order to use that specific data later for study purposes. For instance, the School Management System must be capable of keeping the marks of the students and distinguishing between the student's grade. One of the main parts of the School Management system is the School Attendance Register, which records the daily student attendance and at the end of the year it specifies



1. The Images of the students must be in the left-hand side of the section.
2. The attendance must be taken in the right section by tick mark (blue color) or crossing signs (red color).
3. The logo should be on the left side of the page.
4. The submission button must be on the outer side of the section.
5. The "SCHOOL ATTENDANCE REGISTRER" text must be written in the bottom of the page.
6. By clicking the submission button, it must guide us back to the home page of the system.

Figure-2: Customer's description about School Attendance Register Design

the actual attendance percentage of the students. While gathering requirements about the “school attendance registration” in a traditional text-based method, the customer’s description could be something like in figure-2.

In order to include the upper requirements attendance design into the system, software engineers might assume the customers asked for something like in figure-3.

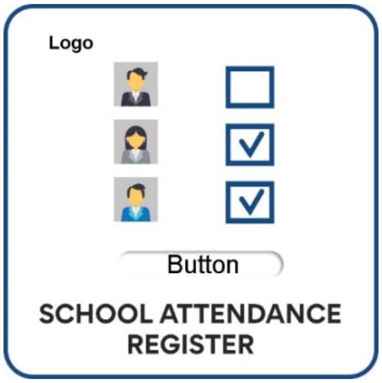


Figure-3: Engineer’s Assumption

However, the actual attendance design that customers want is shown in figure-4.

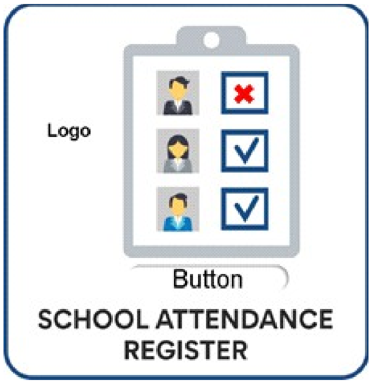


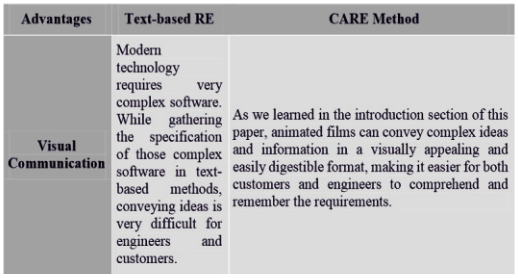
Figure-4: Actual Customer Need

The sixth requirement says, after submitting the attendance, the system must guide us back to the home page. In this proposed method, all these animations must be made by the graphic designers in a cartoon form in order to display the actual system to customers and team members, so they can take their final decision about the required system.

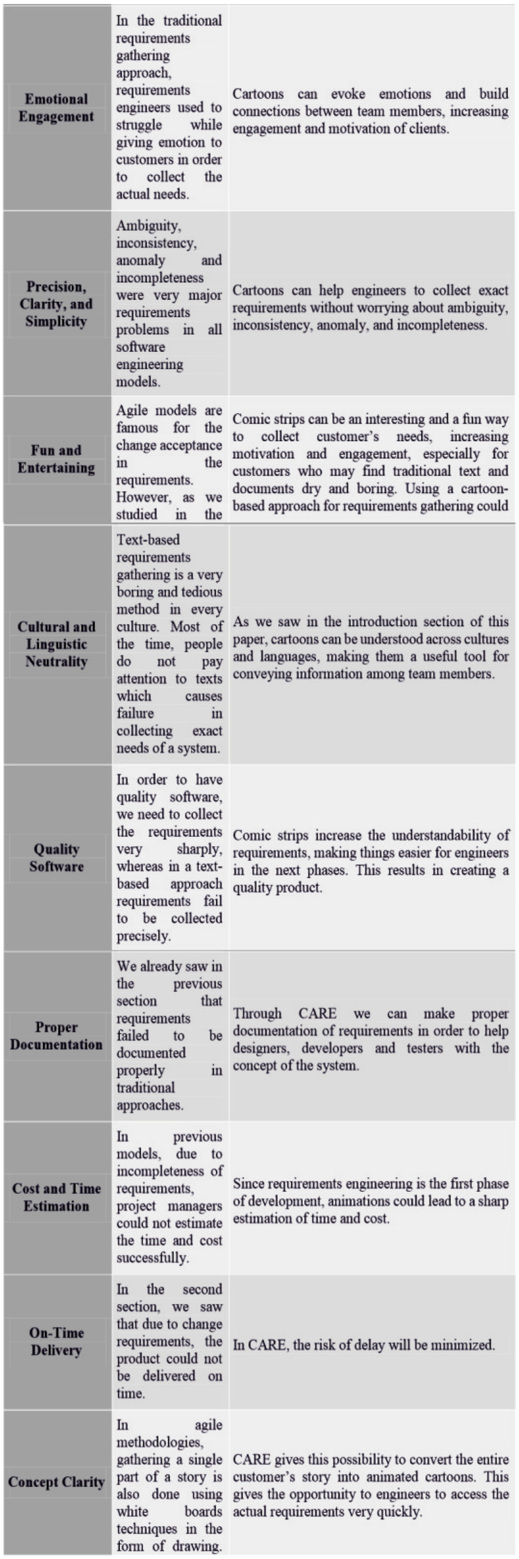
Considering the aforementioned example, the requirements could be modified as soon as the cartoon is made by the graphic designers, and on top of all, if CARE method is used in defining the whole system, it will make the actual system needs much more understandable and better from both customers and engineers. Furthermore, the traditional text-based method does not seem to be clear enough for both the customers and the engineers, whereas CARE provides an entertaining and effective way to understand the software requirements. Additionally, if we create an animated film in a cartoon style to depict the specific requirements, it will look more precise and lucid. It also provides an overall concept to the client to help engineers bring changes to the system. On the other hand, since animation is made in a matter of minutes, creating an animated story in the form of a cartoon brings an overall concept of real user interface. Whereas a real interface cannot be created as quickly as a cartoon. This leads to a very precise requirements gathering process.

There are several advantages to using cartoons in software requirements engineering, as opposed to traditional text and documents, listed in the following table:

In addition to the benefits mentioned above, there are



Advantages	Text-based RE	CARE Method
Visual Communication	Modern technology requires very complex software. While gathering the specification of those complex software in text-based methods, conveying ideas is very difficult for engineers and customers.	As we learned in the introduction section of this paper, animated films can convey complex ideas and information in a visually appealing and easily digestible format, making it easier for both customers and engineers to comprehend and remember the requirements.



Emotional Engagement	In the traditional requirements gathering approach, requirements engineers used to struggle while giving emotion to customers in order to collect the actual needs.	Cartoons can evoke emotions and build connections between team members, increasing engagement and motivation of clients.
Precision, Clarity, and Simplicity	Ambiguity, inconsistency, anomaly and incompleteness were very major requirements problems in all software engineering models.	Cartoons can help engineers to collect exact requirements without worrying about ambiguity, inconsistency, anomaly, and incompleteness.
Fun and Entertaining	Agile models are famous for the change acceptance in the requirements. However, as we studied in the	Comic strips can be an interesting and a fun way to collect customer's needs, increasing motivation and engagement, especially for customers who may find traditional text and documents dry and boring. Using a cartoon-based approach for requirements gathering could
Cultural and Linguistic Neutrality	Text-based requirements gathering is a very boring and tedious method in every culture. Most of the time, people do not pay attention to texts which causes failure in collecting exact needs of a system.	As we saw in the introduction section of this paper, cartoons can be understood across cultures and languages, making them a useful tool for conveying information among team members.
Quality Software	In order to have quality software, we need to collect the requirements very sharply, whereas in a text-based approach requirements fail to be collected precisely.	Comic strips increase the understandability of requirements, making things easier for engineers in the next phases. This results in creating a quality product.
Proper Documentation	We already saw in the previous section that requirements failed to be documented properly in traditional approaches.	Through CARE we can make proper documentation of requirements in order to help designers, developers and testers with the concept of the system.
Cost and Time Estimation	In previous models, due to incompleteness of requirements, project managers could not estimate the time and cost successfully.	Since requirements engineering is the first phase of development, animations could lead to a sharp estimation of time and cost.
On-Time Delivery	In the second section, we saw that due to change requirements, the product could not be delivered on time.	In CARE, the risk of delay will be minimized.
Concept Clarity	In agile methodologies, gathering a single part of a story is also done using white boards techniques in the form of drawing.	CARE gives this possibility to convert the entire customer's story into animated cartoons. This gives the opportunity to engineers to access the actual requirements very quickly.

Table-1: Text-based vs CARE

some potential challenges associated with using cartoons for software requirements engineering:

- **Excessive involvement:** In this style, too much face-to-face communication could be wearisome. We cannot force the customers to stay and accompany us all the time, since there is a saying: "Customer is King". On the other hand, other team members such as designers, developers, testers also need to stay with us all the time, this could cause loss of energy.
- **Lack of Formality:** The use of cartoons might make the requirements engineering process less formal, which can lead to a serious lack of rigor and attention to detail.
- **High Salaries of Graphic Designers:** Nowadays, the salaries of graphic designers are very high. This could increase the cost of the project by 1 to 2 percent.
- **Experienced Team:** In this approach, the focus is on requirements gathering and analysis. In order to collect the exact customer's needs and convert it into motion-graphics, we need a highly experienced and skilled team.

While using cartoons for engineering requirements can have significant benefits, it is important to carefully consider the specific needs and limitations of the project and the team before deciding to use this approach.

4 Conclusion

In the software development life cycle, the requirements gathering and analysis phase lays the groundwork for the software. Traditional styles of requirements gathering were often blurred, ambiguous, and time-consuming, making it challenging for software engineers to collect customer's needs. However, cartoons are a powerful tool that can minify and reduce the risk of misunderstandings and miscommunications. The Cartoon-assisted Requirements Engineering (CARE) method is a new and unique approach to requirements gathering that leverages the power of visual communication to ensure software requirements are well-defined, clearly understood, and accurately represented.

One such tool is DALL-E, a deep learning model developed by OpenAI. DALL-E is a variant of the GPT (Generative Pre-trained Transformer) model that is trained to generate images from textual descriptions. By feeding a textual description into DALL-E, the model generates a corresponding image, which can be a cartoon or any other type of image.

Future Work

Artificial intelligence (AI) has brought many facilities and advancements in a wide range of fields and industries, such as healthcare, transportation, finance, education, and more. AI is a rapidly evolving field, and new applications and advancements are being discovered all the time. As AI continues to advance, it is likely that it will bring even more facilities and benefits in a wider range of fields, disciplines and industries.

The ability to create precise cartoons from text has advanced significantly in recent years, thanks to the development of AI. It is possible to create precise cartoons from text in today's world using numerous tools and techniques. However, more advancements are still required in this field to create an exact cartoon from a text-based story, which can be used in CARE form.

References :

- Mall, R. (2018). Fundamentals of software engineering. PHI Learning Pvt. Ltd..
- Alshamrani, A., & Bahattab, A. (2015). A comparison between three SDLC models: waterfall model, spiral model, and Incremental/Iterative model. International Journal of Computer Science Issues (IJCSI), 12(1), 106.
- Balaji, S., & Murugaiyan, M. S. (2012). Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. International Journal of Information Technology and Business Management, 2(1), 26-30.
- Sharma, S., Sarkar, D., & Gupta, D. (2012). Agile processes and methodologies: A conceptual study. International journal on computer science and Engineering, 4(5), 892.
- Scacchi, Walt. (2002). Process Models in Software Engineering. 10.1002/0471028959.sof250.
- Rengur, Z., & Sugirin, S. (2019, July). The effectiveness of using comic Strips to increase students' reading comprehension for the eighth-grade students of SMPN 1 Pundong. In 6th International Conference on Educational Research and Innovation (ICERI 2018) (pp. 511-515). Atlantis Press.
- Sommerville, I. (2011). Software engineering 9th Edition. ISBN-10, 137035152, 18.
- Anida, N. (2019). The Use of Comic Strips to Improve the Students' Reading Comprehension of Narrative Text (An Action Research at The Seventh Grade Students of MTsN 3 Medan in Academic Year 2019/2020) (Doctoral dissertation, Universitas Islam Negeri Sumatera Utara).
- Sabale, R. G., & Dani, A. R. (2012). Comparative study of prototype model for software engineering with system development life cycle. IOSR Journal of Engineering, 2(7), 21-24.
- Pohl, K. (2010). Requirements engineering: fundamentals, principles, and techniques. Springer

Publishing Company, Incorporated.

- McGowan, D. (2019). Animated Personalities: Cartoon Characters and Stardom in American Theatrical Shorts. University of Texas Press.
- Dobson, N. (2020). Historical Dictionary of Animation and Cartoons. Rowman & Littlefield Publishers.
- Dobson, N. (2010). The A to Z of Animation and Cartoons (No. 194). Rowman & Littlefield.
- Oliveira, N., Costa, J., Silva, C., & Ribeiro, B. (2020). Retweet predictive model for predicting the popularity of tweets. In Proceedings of the Tenth International Conference on Soft Computing and Pattern Recognition (SoCPaR 2018) 10 (pp. 185-193). Springer International Publishing.
- Streeten, N. (2020). UK Feminist Cartoons and Comics. Springer International Publishing.
- Wiley, J. (2019). Picture this! Effects of photographs, diagrams, animations, and sketching on learning and beliefs about learning from a geoscience text. Applied Cognitive Psychology, 33(1), 9-19.