Component-Driven Development in Modern Virtual Assistants: A Mapping Study

Reza M. Parizi[1*], Ahmad Shahi[2]

[1] Department of Software Engineering and Game Development, Kennesaw State University, Marietta, GA 30060, USA.
[2] Department of Information Science, University of Otago, Dunedin 9016, New Zealand.

* Corresponding author. Tel.: +1-470-578-2118; email: rparizi1@kennesaw.edu

Abstract: Virtual assistants are becoming more common today as access to this technology has become cheaper and more affordable to the common consumer. As a result of this increase in demand, a component based software engineering approach may be more practical in the future development of these systems. This article presents a study of the current state of the research on component based software engineering and service principles in virtual assistants. The main motivation for this research is that there just is not any comparative study on how component-driven development might benefit our virtual assistants, especially those with voice interaction. This study is limited to papers that go into detail about the development of intelligent personal assistants and the entities that comprise them. Due to the nature of virtual assistants, we initially assumed that many of them would primarily be made of various software components. However, our results show that the development of virtual assistants is actually more complex than we initially believed, and that a more consolidated approach is necessary to define the best method of development for virtual assistants.

Key words: Component, intelligent agent, intelligent personal assistant, multi-agent system, software agent, virtual assistant.

1. Introduction

As virtual assistants are growing in popularity, there is also a growing need for a unified development process for virtual assistants. Virtual assistants are a form of software agent that performs tasks or services for a user. The capabilities of virtual assistants have been rapidly expanding as technological advancements are made. Beginning in the year 1961, the IBM shoebox was unveiled, becoming the first virtual assistant to be capable of solving basic math problems in response to voice commands [1]. Eventually, as more technological advancements were made, in the year 2011 Apple would release the iPhone 4S, which hosted the first modern intelligent personal assistant, Siri. Intelligent personal assistants are a highly advanced form of virtual assistants, capable of performing a number of tasks on behalf of their users. For example, performing internet queries, various tasks of automation, and even interacting with Internet-of-Things (IoT) devices. As of the year 2017, advancements in the development of virtual assistants have allowed for the creation of more non-commercial virtual assistant applications. The current application areas of virtual assistants range from open-source virtual assistants such as Sirius, to library assistants like Byblon [2], and finally advanced intelligent personal assistants such as Amazon Alexa.

Fig. 1. Examples of modern intelligent personal assistants [3].

Currently, there is not a well-defined standard for developing modern virtual assistants. Component Based Software Engineering (CBSE) is one such possible development method that could be used. CBSE is made up of two general processes, development with reuse and for reuse. The process of developing with reuse can be defined as utilizing previously created software components and services in the development of new software applications. The other process, development for reuse, is concerned with developing components or services that will be reused in other applications.

The benefits to utilizing CBSE in the development of virtual assistants include the ability to create and reuse software components, making future development more simplified. The goal of this paper is to conduct a systematic mapping of the current state of research of the use of CBSE in the development of modern virtual assistants since virtual assistants in recent years are becoming more and more integrated into our daily lives, like in systems such as IoT.

The remainder of the paper is structured as follows: Section 2 presents the formulation of research questions and the search query. Section 3 discusses the methodology of the systematic mapping that was conducted, and the complications of the research that could threaten its validity. Section 4 discusses our findings and results of our research. Section 5 presents related work. Finally, Section 6 provides conclusions that could be drawn from study with regards to CBSE within virtual assistants, as well as plans to move forward with this research.

2. Research Questions and Search Query

The process that the team followed to plan the research of the systematic mapping includes the creation of research questions, the motivations behind our research questions, the selection of key terms, the creation of the search string, a constraining of available sources, a method of data extraction, and finally an analysis of the extracted data. The main question that the research is intended to answer is:

How might a component based approach affect the development of virtual assistants?

In order to better answer this question, and understand the applications of CBSE within the development of virtual assistants, we created three more questions to facilitate this research (see Table 1). Each question was written with the express purpose of identifying the previous application areas of components within virtual assistants, their application within the future development of virtual assistants, and finally identifying alternative methods of development for virtual assistants that could be either superior to CBSE, or perhaps used in conjunction with CBSE.

The key terms for this research were selected based on what was perceived to be relevant to the presence of CBSE within the development of virtual assistants. As such, the most general terms chosen were "component" and "virtual assistant." Due to the nature of virtual assistants varying in their naming and application, the terms "software agent," "multi-agent system", "intelligent agent", and "intelligent personal assistant" would later be added. Due to the rise of modern virtual assistants such as Siri and other intelligent personal assistants, more terms were necessary to broaden the coverage of different virtual assistants.

Table 1. Research Questions

Research Questions	Motivation for Research
RQ1. How might a component based approach affect the development of virtual assistants?	To identify the advantages or disadvantages of using CBSE in the development of virtual assistants, in order to prove or disprove our hypothesis that CBSE is the preferred development method.
RQ2. What have the application areas of component based software engineering been in the development of virtual assistants?	To determine the application areas of CBSE in the development of virtual assistants.
RQ3. What future works did authors propose on the development of virtual assistants with component based software engineering?	To identify the uses of components within proposed virtual assistants.
RQ4. What alternative development methods are being used for virtual assistants outside of component based software engineering?	To identify and compare software development methods for the creation of virtual assistants.

3. Methodology

3.1. Conduction of the Systematic Mapping

The study was carried out by first defining our research questions, applying various constraints to our search, removing duplicates of research papers, removing non-relevant research papers based on abstract, and later full text, extracting data from the final list of relevant sources, and finally mapping and drawing information from this data. After the planning of the research phase, the initial query was inputted within two different databases. The two databases chosen were the Kennesaw State University library database (which covers publications in all major digital libraries such IEEE Xplore, ACM, Springer, Wiley etc.) and the Google Scholar database. The aforementioned databases were assumed to contain relevant and valid resources for this research topic.

First, the selected key terms were entered into these two databases, and an initial pull of papers was retrieved.   There was a total of 494 results from the Kennesaw State University Library collection and 666 Results from the Google Scholar database. The Kennesaw State University Library collection search was given these initial constraints:

- Available within the Library Collection
- Full-text
- Scholarly Peer Reviewed Journals
- English
- Published from 2018 and earlier

For the Google Scholar database, the initial constraints were slightly different as a result of the available constraining options. Instead of applying the exact same constraints, we instead utilized similar constraints:

- English
- Published from 2018 and earlier

After applying these constraints, the number of results obtained from the Kennesaw State University Library collection was reduced to 132, as shown in Fig. 2, while the results from the Google Scholar database

was reduced to 395, as shown in Fig. 3. The total number of results at this stage, was 526.
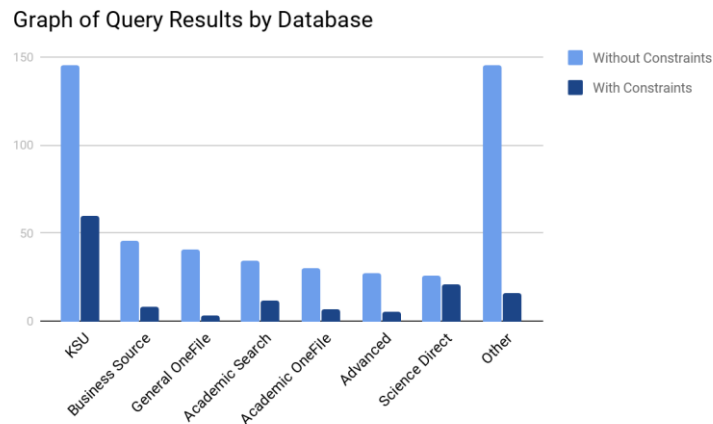
Graph of Query Results by Database

Fig. 2. Graph comparing the query results for KSU's database with and without the initial constraint.
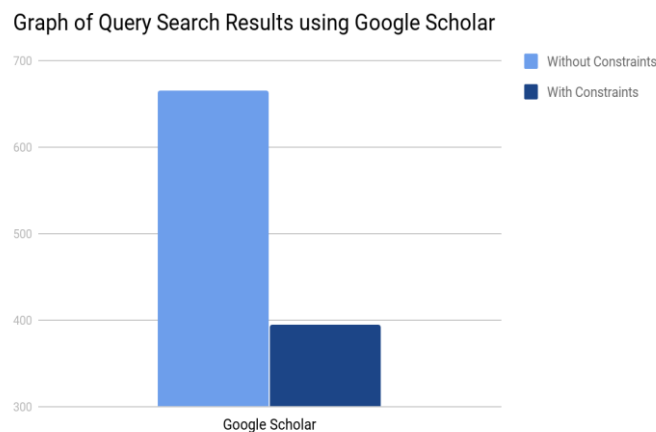
Graph of Query Search Results using Google Scholar

Fig. 3. Graph comparing the query results for Google Scholar's database with and without the initial constraint.

From this point forward, a publishing date constraint was applied, which limited our results to only those written from the year 2011 and onward. This date was chosen, because it is the date in which virtual assistants were revolutionized due to Apple's iPhone 4S being released. The release of this device introduced the intelligent personal assistant, Siri. Around this time, similar virtual assistants were either being developed or proposed. We hypothesized that adding this constraint to dates would allow for us to acquire more relevant information regarding modern virtual assistants. After adding this constraint, we had a total of 266 results.

With the remaining papers, we compared the search results from all databases and removed all present duplicates, resulting a total of 264 results. Next, the team began qualitative analysis of the papers. At this stage, we first looked at all of the abstracts of the remaining research papers to determine their relevance to our study. Any paper that did not answer a research question was deemed irrelevant and removed from our pool of documents. The result of this removal process left us with 79 research papers. Finally, the team went through each remaining research paper, and read each one fully, and extracted relevant information. Papers that did not contain relevant information, were removed from our final pull of documents. After removing all irrelevant papers, the final pool of relevant research papers totals at 31 (as shown in Fig. 4 and Fig. 5).
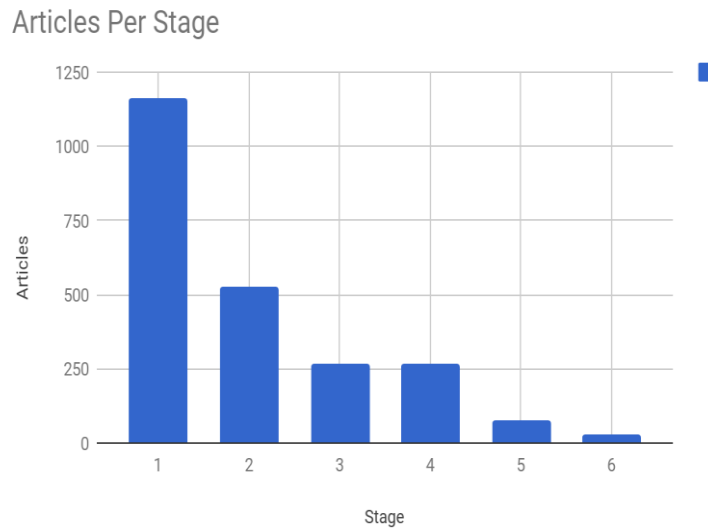
Articles Per Stage

Fig. 4.　Articles per stage of the systematic mapping process.

| Apply Search Query to Google Scholar and KSU Library Databases | Applied Initial Constraints | Applied time Constraint | Remove duplicates from initial search | Removed all Non-Relevant papers based on abstract | Removed all Non-Relevant papers based on full text |

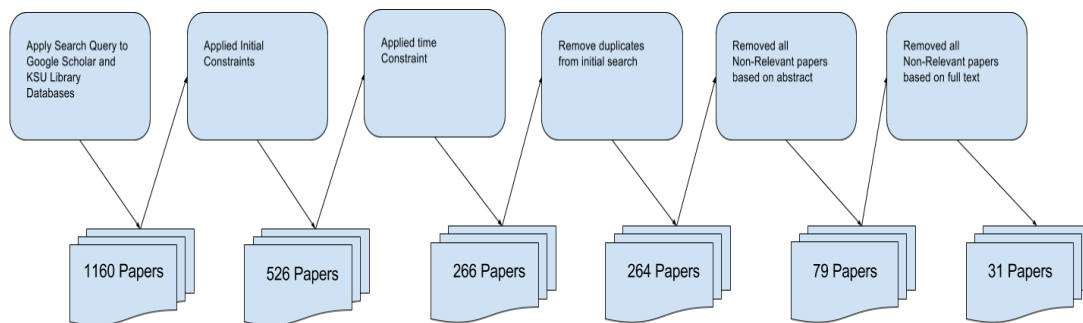| 1160 Papers | 526 Papers | 266 Papers | 264 Papers | 79 Papers | 31 Papers |

Fig. 5. Systematic mapping study source selection process.

During the systematic mapping process, we analyzed the final pool of research papers for various statistics. One such statistic that we found relevant, was the number of research papers per publication year. Beginning in the year 2011, the number of published papers relevant to the topic of CBSE in virtual assistants has seen a steady increase over time. One could hypothesize that the introduction of Apple's Siri in 2011, and the subsequent releases of other commercial virtual assistants, sparked an interest in this topic, thus increasing the number of publications each year (as shown in Fig. 6).

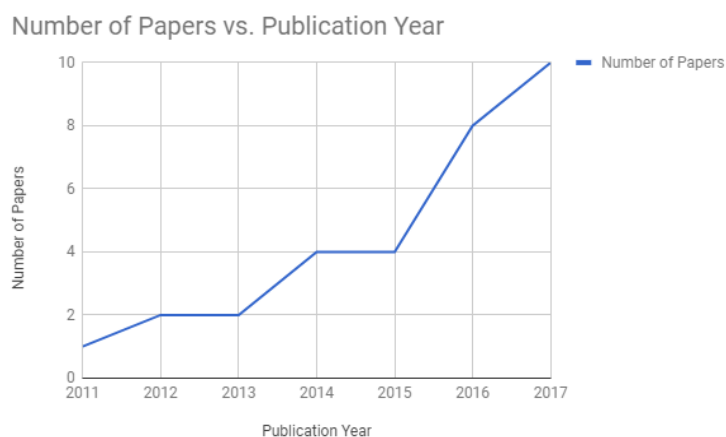Number of Papers vs. Publication Year

Fig. 6. A line graph showing publication years, and the number of papers relevant to the study.

One other statistic on the distribution of articles was sought out during the research. This statistic was found due to the team's notice of many of our removed resources coming from the Kennesaw State University library collection. Within Fig. 7, we found that many of the results were coming from Google Scholar's database.
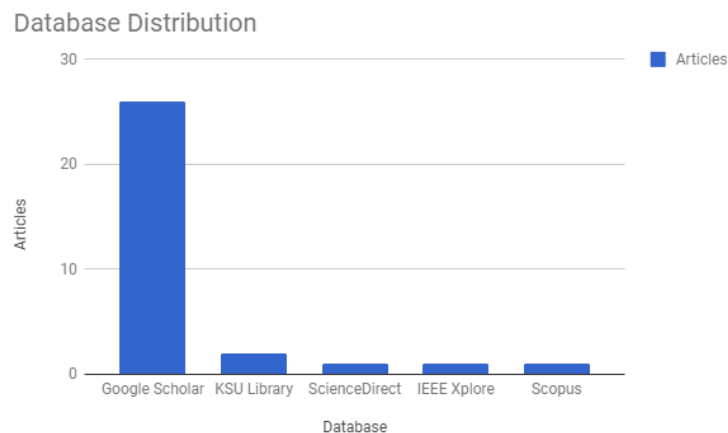

Fig. 7. Number of articles per database.

3.2. Threats to Validity of the Study

Throughout our research, we came across multiple problems that made us rethink the approach to our study. This included reworking our research questions, dealing with the presence of technical terms in academic papers being used interchangeably, and our initial search query being too narrow.

As we conducted our initial research and analysis of our article pool, we came to the realization that there was not much research on whether CBSE is an effective development method for virtual assistants. Our research did provide us with articles showing the use of CBSE in virtual assistants, however, we could not use these articles to answer one of the initial research questions. This research question:

How has component based software engineering in intelligent personal assistants been researched in the past?

Was then removed from our research due to lack of data. Thus we decided to conclude that given our research, there are no articles containing research on the topic of CBSE in the development of virtual assistants.

During the course of our research, we encountered many instances of the misuse of technical terms in multiple papers. The first instance of this was authors using technical terms such as component or agents, but the way they used the term was inaccurate. This caused an issue of validity, since the papers that we assumed to be relevant turned out to be irrelevant, detracting from our pool of useful information. We could not draw any conclusions from these papers, since the information was not reliable. The second instance of this was when authors created their own terminology for concepts that were already defined in the academic space. An example of this would be an article that uses the term "ubiquitous personal assistant" instead of using the proper term, virtual assistant [4]. This made it difficult to determine which development methods the paper was referring to.

Our initial query was formed based on our collective knowledge of CBSE and virtual assistants. Due to this, certain terms such as "services" or other applications of CBSE could not be used in our query. Midway through our research, we discovered that there are articles relevant to our study outside our initial article pool that referred to virtual assistants with terminology that was not used in our initial query. Some of these articles also detailed development methods for virtual assistants outside of CBSE. This means that there could

have been relevant information that we were unable to draw conclusions from. At this point in our research, it was too late to redo the search query to include these papers. In order to maintain validity, the decision was made to continue research without reworking our search query, and to draw our conclusions based solely on the articles from our pool.

4. Results and Discussion

This section will facilitate discussion and provide answers to each of the research questions shown in Table 1.

4.1. RQ1. How Might a Component Based Approach Affect the Development of Virtual Assistants?

Of the remaining 31 papers, only 5 of them (16.129%) actually discussed the usage of components within this research, the CBSE processes, and their effects on the development of virtual assistants. Within one document, an intelligent virtual assistant was being developed by a team of researchers. The teams of researchers were utilizing open source components, which is an example of the CBSE process of "development with reuse." Three different open source components were selected for the development of this intelligent virtual assistant. A component was selected to manage text annotations, question answering, and information extraction [5]. In this scenario, the ability to reuse components in the development of virtual assistants can greatly decrease the amount of time necessary for development. One other example from the pool of research papers discussed the building of a neurological framework that utilized both the "development for reuse" and "development with reuse" CBSE processes. The virtual assistant discussed within this research paper talks specifically about developing components for the framework, while also utilizing external components to help with adding additional features and accommodating external functionalities [6]. Other examples include the *Ubiquitous Personal Assistant* (UPA) discussed by Cristina Mihale-Wilson, and its potential for "development with reuse" due to the purchased components of or for the UPA and E-VOX, a virtual assistant designed to search information from Wikipedia, which utilizes the CBSE process of "development for reuse," and goes into the specifics of each component to be developed for the virtual assistant [4], [7].

The main disadvantage with using CBSE in the development of virtual assistants is that it is difficult to ensure the security of the components. One of the main advantages of CBSE is that you can generalize your components in order to reuse them in multiple different applications. However, when you generalize like this, the communications between the components also become generalized. This makes it more difficult to secure these connections, making any private or important data being communicated more vulnerable to exploitation. This is even more concerning since virtual assistants in recent years are becoming more and more integrated into our daily lives, like in systems such as IoT. If the data from the figure 5 is indicative of a trend of the popularity of virtual assistants, then we can extrapolate that virtual assistants will become commonly used by the public in the near future. It would then be advantageous for developers of virtual assistants to carefully consider the security aspect of developing these systems in order to ensure the overall safety of the public [8].

4.2. RQ2. What Have the Application Areas of Component Based Software Engineering Been in the Development of Virtual Assistants?

Component based software engineering has been applied in various areas where virtual assistants are implemented. Areas like health, education, query engines, navigation modelling, repositories and automation have greatly benefited from component based software engineering through the development of virtual machines (as shown in Figure 8).
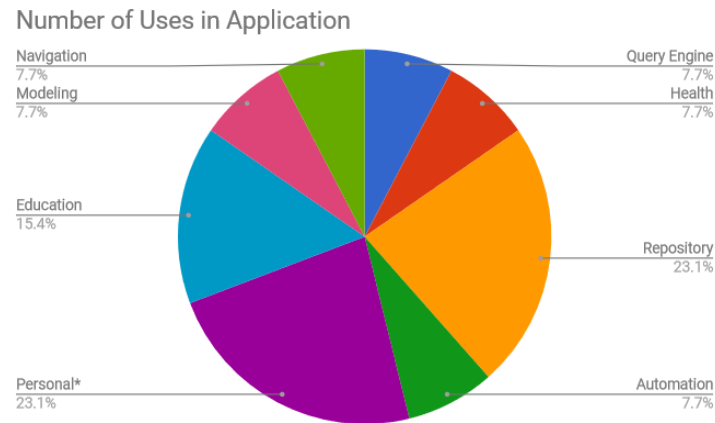
Number of Uses in Application

Fig. 8. Different application areas of components in virtual assistants.

A software agent called Automated Nursing Agent (ANA) has tremendously improved how family members take care of their elderly loved ones by using the at-home elderly care methodology. The ANA is a software agent that runs on an android tablet device and is specifically designed for the companionship of the elderly [9]. It verbally communicates to the patients/elderly person and provides with companionship.

Unlike intelligent personal assistant software agents such as Apple Siri, Google Now, Microsoft Cortana, and many others, which are designed to answer information requests or to help owners of handheld devices to verbally express commands on their devices, ANA piggybacks on Google APIs for information requests but extends this capability by building a personalized knowledge base to provide personalized conversations with answers but also questions and general statements [9].

Parental Affective Natural Driving Assistant (PANDA) is another virtual assistant that parents with young kids have started using in order to concentrate on driving as their kids also get the attention they need. PANDA provides a tool that can entertain, educate and engage the children while in the car on a trip.

PANDA is designed to address the following issues:

- Parental assistance in reducing attentional and task load.
- Parent-supervised entertainment for children in the car.
- Social, educational and relational activities.

To fulfill these goals, PANDA has a face that appears on either the front- or back- seat tablets, can produce facial expressions, recognize speech and interact with the driver and passengers in the car in a socially engaging way [10].

One application of CBSE in virtual assistants utilizes modern statistical machine learning to be applied in a language learning environment. The system stresses the use of CBSE by designing with modularity to ensure scalability. The result is a tool that properly detects user intent in a voice driven virtual assistant. This system tackles many of the issues developers of voice driven virtual assistants face, such as properly determining user intent, speech to text processing, and supporting multiple languages [11].

In one article gathered during our research, the development process of virtual assistants is analyzed and different frameworks for developing these systems are compared. Two such frameworks are ontology-based systems and data-driven systems based on partially observable Markov decision process. Both of these approaches deal with different degrees of human interaction and automation, where both have their tradeoffs [12].

LISSA is a personal assistant designed to operate in an education IoT ecosystem. The purpose of this assistant is to provide an access point to the IoT environment, analysis and recommendations based on the

user, and other common functions of modern virtual assistants. The system is made up of two main components, a Generic Personal Assistant and a Generic Dispatcher Assistant [13].

In [14], a proposed application, Mobile Augmented Reality in Archaeology (MARA), is characterized by providing proactive and autonomous suggestions within an augmented reality context. Specifically, the system is designed to provide additional information for various archaeological sites and function as a type of virtual assistant. It is comprised of 5 different components, each working together to essentially act as an intelligent personal assistant tour guide. The first component, is designed to present an Environment map which imports a GPX trace for points of interest. The second component is utilized to describe these points and sites of interest and to annotate related media. The third component is a digital repository, which stores digital objects pertaining to the points of interest. The fourth component, is responsible for comparing images and other media, as it is able to read the content descriptor for every digital object. Finally, the last component is an extended version of an open source navigator, known as Navit. This component provides a sort of GPS for the system.

Sirius is the first non-commercial open source virtual assistant. The system utilizes CBSE in order to handle speech recognition, natural language processing, and computer vision. For the question answering component, Sirius makes use of the OpenEphyra framework developed from Carnegie Mellon University's collaboration with IBM. The automatic speech recognition component is made up of a Gaussian mixture model framework called Sphinx. Finally, there is also an image-matching pipeline component that communicates with a processed image database and analyzes images using a SURF algorithm [15].

The applications of CBSE within virtual assistants vary in use; however, some of the most popular uses for components within them, have been in personal uses, which is defined by components used in virtual assistants that aid users in managing daily activities (e.g. commercial virtual assistants like Siri or Cortana). One of the other primary uses of components within virtual assistants have been to serve as a type of data item repository. These components are responsible for holding and managing different types of data (e.g. MARA's data repository [14]).

4.3. RQ3. What Future Works Did Authors Propose on the Development of Virtual Assistants with Component Based Software Engineering?

Several works utilizing virtual assistants with component based software engineering have been proposed to be incorporated in a variety of areas. These works cover an internet of things (IoT) architecture utilizing intelligent personal assistants (IPA), and various IPA proposals.

One example proposes a software and hardware combination that uses social networking platforms as an interface for automation in a home or building. The architecture uses a bot through a desired social network (Facebook in its example), an interface that processes commands and responses, and a physical layer that controls the desired automation. The application used in the bot layer is known as the Robin Bot. The Robin Bot functions as a Facebook account that is linked to the rest of the service. By starting a conversation with the bot, the bot links the user's ID to their Facebook ID and MAC address, allowing the requests made by the user to be incorporated in the correct appliance [16].

Other future works proposed a virtual mobile assistant designed specifically for speakers of the Vietnamese language and an intelligent personal assistant for everyday use and management of daily activities [17], [18]. The application, named VAV, while similar to other applications, it is the first to utilize smart language analysis and understanding designed specifically for the Vietnamese. The components of VAV would be similar to the other virtual assistants, with applications relating to a component and actions for that component to take [18].

One other intelligent personal assistant proposed would utilize a server component for managing requests for interactions with objects within an IoT architecture [17]. This proposed work is yet another example of

the applications of CBSE in the realm of virtual assistant applications.

Another future work proposed is a Smart Classroom Framework (SCF) that uses a context-sensitive form of a IPA to be used on a mobile device to assist students inside and outside the classroom. SCF consists of four main components: SCCentral, SCHub, SCApps, and SCStudentApp. SCCentral runs on the central server that the other components relate back to, typically on the a server with a known IP address. SCHub is run on individual classroom servers, registering with SSCentral through HTTP protocol. SCApps are packages installed on SCHub, SCApps would be available for download from an SCApps repository. These apps would be accessed by the students via the SCStudentApp that would be installed on their personal device [19].

The data collected from the proposed works seem to indicate that Component Based Software Engineering is still being used within future projects and in various new application areas

4.4. RQ4. What Alternative Development Methods are Being Used for Virtual Assistants Outside of Component Based Software Engineering?

IPAs can be developed through a variety of methods, not just a component based approach. Some of the other methods utilized in the development of virtual assistants include Agent-oriented Software Engineering (AOSE).

AOSE is another development approach to developing multi-agent systems, which focuses on the use of agents and organizations of agents. Additionally, agents are best utilized in applications that involve the communication between components [20]. Due to the nature of virtual assistants relying on multiple software components, this method of development may often be used in conjunction with Component Based Software Engineering in virtual assistants.

One example of the AOSE approach and its implementation in a virtual assistant, is within the Multi-purpose Intelligent Assistant proposed in [21]. The assistant utilizes an agent architecture. This example is one of the many examples found of virtual assistants implementing the AOSE approach.

5. Related Work

The research given in this paper is related to secondary studies whose aim is centred on the development of virtual assistants. To the best of our knowledge, there is literally no specific work in the literature that has expanded considerably in recent years in this area. In this sense, our work could be seen as the first step towards facilitating and empowering future research in this domain.

Of the very limited work, Popchev and Ivanov [22] conducted a study to "find the scientometric dependencies" within the research of Intelligent Personal Assistants (IPA). They constrained their search to the twelve years preceding the year 2016. In this study, the two researchers identified 10 relevant keywords for their research. They were able to draw conclusions relating to this field, such as the fact that research into a type of virtual assistant, intelligent personal assistants, is still relatively low when compared to research within similar topics such as Artificial Intelligence. Their most relevant conclusion, is that research in this field has not actually been explored in great detail before the year 2011; however, after this year, there began a rapid growth of interest in this field.

6. Conclusion and Future Work

Virtual assistants are growing in popularity, there is also a growing need for a unified development process for virtual assistants. Currently, there is not a well-defined standard for developing modern virtual assistants. We conjectured Component Based Software Engineering (CBSE) could be one such possible development method that might be used due to the popularity of services and microservices. Based on the results from our systematic mapping, we can conclude that component based software engineering is a viable method for developing virtual assistants; however, many of the virtual assistants mentioned in our pool of articles are

utilizing components in conjunction with additional development methods, such as Agent-oriented Software Engineering, or Service-oriented Software Engineering.

In the future, we intend to conduct a supplemental systematic study in order to expand the work presented in this systematic mapping beyond component-driven software engineering. Such systematic mapping should be more generic, covering all possible development methods currently in use for creating virtual assistants (either commercial or open-source). This study should also attempt to collect and document all alternative terms for virtual assistants, so that it can assuredly say one way or another, which development methods have what benefits for the creation of virtual assistants in order to determine which method is most suitable.

Acknowledgment

References

[1]   Hernandez, D. (2013). Tech time warp of the week: IBM speech recognition.

[2]   Tramullas, J., & Garrido, P. (2012). Library automation and OPAC 2.0: Information access and services in the 2.0 landscape: Information access and services in the 2.0 landscape.

[3]   Anthony, T. (2016). Will intelligent personal assistants replace websites?

[4]   Mihale-Wilson, C., Zibuschka, J., & Hinz, O. (2017). About user preferences and willingness to pay for a secure and privacy protective ubiquitous personal assistant. *Proceedings of the 25th European Conference on Information Systems (ECIS)*.

[5]   Lamfontagne, L., Laviolette, F., Khoury, R., & Bergeron-Guyard, A. (2014). A framework for building adaptive intelligent virtual assistants. *Proceedings of the IASTED International Conference on Artificial Intelligence and Applications*.

[6]   Cingillioglu, N. (2017). Neural logic framework for digital assistants. Department of Computer Science, Imperial College London.

[7]   Pérez, J., Cerezo, E., & Serón, F. (2016). E-VOX: A socially enhanced semantic ECA. *Proceedings of the International Workshop on Social Learning and Multimodal Interaction for Designing Artificial Agents*.

[8]   Chung, H., Iorga, M., Voas, J., & Lee, S. (2017).   Alexa, can I trust you? *Computer, 50*, 100-104.

[9]   Dinevari, A., & Zaiane, O. (2016). Automated nursing agent: A software agent for at-home elderly care. *Proceedings of the Eighth International Conference on eHealth, Telemedicine, and Social Medicine*.

[10]   Gordon, M., & Breazeal, C. (2015). Designing a virtual assistant for in-car child entertainment. *Proceedings of the 14th International Conference on Interaction Design and Children*.

[11]   Burges, C., Hart, T., Yang, Z., Cucerzan, S., White, R., Pastusiak, A., & Lewis, J. (2016). A base camp for scaling AI.

[12]   Bellegarda, J. (2013). Large–scale personal assistant technology deployment: The Siri experience. *Proceedings of the Annual Conference of the International Speech Communication Association*.

[13]   Todorov, J., Valkanov, V., & Popchev, I. (2017). Intelligent personal assistant for aiding students. *Proceedings of the International Conference "Automatics and Informatics'2017*.

[14]   Amato, A., Di Martino, & B. Venticinque, S. (2012). Semantically augmented exploitation of pervasive environments by intelligent agents. *Proceedings of the IEEE 10th International Symposium on Parallel and Distributed Processing with Applications*.

[15]   Hauswald, J., Laurenzano, M., Zhang, Y., Li, C., Rovinski, A., Khurana, A., Dreslinski, R., Mudge, T., Petrucci, V., Tang, L., & Mars, J. (2016). Sirius implications for future warehouse-scale computers. *IEEE Micro*.

[16]   Basaca-Preciado, L., Moreno-Partida, A., Terrazas-Gaynor, J., Ponce, M., Lopez, J., Rodriguez-Quinonez, J., Fuentes, W., & Sergiyenko, O. (2017). Home and building automation through social networks.

Proceedings of the 2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe).

[17] Santos, P., Casal, J., Santos, J., Varandas, L., Alves, T., Romeiro, C., & Lourenco, S. (2015). A pervasive system architecture for smart environments in internet of things context. *Proceedings of the 17th International Conference on Multimodal Interaction.*

[18] V. T. H, Y., P. T, T., T.Q, T., N.T, T., D. Q, V., & N. H. V., (2016). On the design and development of a mobile virtual assistant for vietnamese(VAV). University of Engineering and Technology-Vietnam National University.

[19] Myat, S. (2014). Enabling context aware applications in learning environments. Master's Thesis, National University of Singapore.

[20] Agentbuilder.com. (2011). Why, when, and where to use software agents.

[21] Konecki, M., Gams, M., & Ivkovic, N. (2015). MIA: A multi-purpose intelligent assistant. *Proceedings of the 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO).*

[22] Popchev, I., & Ivanov, A. (2016). Scientometric dependencies in the field of intelligent personal assistants. 50-16.

Reza M. Parizi is a software engineering faculty in the Department of Software Engineering and Game Development at Kennesaw State University, GA, USA. He is a consummate technologist and researcher with an entrepreneurial spirit. He has applied his insights and expertise to a host of innovative and technology driven projects across start-ups, IT, retail and education industries. He received a Ph.D. in Software Engineering in 2012 and M.Sc. and B.Sc. degrees in Software Engineering and Computer Science respectively in 2008 and 2005. He has more than 8 years of working experience in industrial software development and project managing.

His interests are R&D in software engineering, service and component engineering, software and mobile testing, gamification, software traceability, IoT programming, software development tools, empirical studies. He has published several research papers in top reputable scientific journals and international conferences and also has two copyrights to his credit.

Ahmad Shahi is a doctoral researcher in the Department of Information Science, University of Otago, New Zealand. His research interests include efficient uncertain data stream clustering algorithm for arbitrary shaped clusters and software engineering. He is serving as an editorial member and reviewer of several international reputed journals. He has authored many research articles/books related to data-driven applications and software engineering.