VIAProMa: An Agile Project Management Framework for Mixed Reality

Benedikt Hensen(✉) and Ralf Klamma

Advanced Community Information Systems Group (ACIS) Chair of Computer Science 5, RWTH Aachen University, Aachen, Germany
{hensen,klamma}@dbis.rwth-aachen.de

Abstract. With the COVID-19 pandemic, distributed and remote working became a necessity but in agile project management, social interactions like daily standup meetings in Scrum are vital for the project success. Mixed reality can provide a new way of combining remote collaboration with innovative 3D visualizations to analyze the project status. In this paper, we present a visual immersive analytics framework for project management (VIAProMa). It imports data from project management tools like the GitHub issue tracker for open-source projects. With these task data as the basis, it can generate three-dimensional visualizations, e.g. about the overall progress or the competences of individual developers. Developers, stakeholders and end users can meet in the collaborative virtual environment as avatars and establish a spatial structure with the task cards and visualizations. Therefore, VIAProMa with its adapted and customized mixed reality project management features supports both the shared meetings and the information flow in the project. The shared environment makes it a suitable tool for DevOpsUseXR, an extension to the DevOps workflow, where end users are able to participate in the development process in mixed reality. The resulting implementation is available as an open-source project with cross-platform capabilities targeting the Microsoft HoloLens, HTC VIVE and Android smartphones, as well as tablets. The framework is applied in university teaching classes to convey agile methodology in mixed reality programming practices.

Keywords: Mixed reality · Project management · Agile methodology

1 Introduction

Agile project management practices like Scrum are commonly used for software development but also non-software projects, e.g. in industry [8]. An important factor for the success of an agile project is social interaction [4]. Even for open-source projects where contributors remain largely anonymous and do not know each other personally, establishing a community is important for the success

L. T. De Paolis et al. (Eds.): AVR 2021, LNCS 12980, pp. 254–272, 2021.
https://doi.org/10.1007/978-3-030-87595-4_19

of the project [4]. Due to the meeting restrictions of the COVID-19 pandemic, these remote settings are not only relevant for the open-source community but for every team. To support the community-building process in such remote settings, a persistent room is required where developers can meet virtually. Within such a virtual room, users should not only be able to steer and control the project milestones and tasks but they should also be able to monitor the progress, as well as the community using immersive analytics [9].

An intuitive solution for such a virtual meeting room is mixed reality. If it is not possible to meet in a real room because of pandemic constrictions but also resource limitations, the virtual room should be as immersive as possible. Here, remote collaborators can be represented by customizable avatars. This enhances the social presence and embodiment by mimicking real world meeting experiences in mixed reality [14]. They can interact with each other and the same content without technological borders [2]. For instance, a discussion can take place in a meeting and participants can generate visualizations in the same environment on the fly in order to strengthen their argumentation. Moreover, mixed reality provides a third dimension which can be utilized by the visualizations to improve the understanding of the viewed data. Another important aspect of successful software projects is the involvement of end users [19] or even the co-design of the application with them [11]. The mixed reality environment provides an ideal platform for this as the end users can join the same developer meetings in order to contribute their opinions and feedback.

Our goal is to investigate whether mixed reality is suitable to support and enhance project management workflows. Furthermore, we aim to identify the challenges and design solutions that are necessary to realize a mixed reality project management framework. In this paper, we present our approach to combine immersive visual analytics with project management. To evaluate the merging of these two concepts, we implemented a visual immersive analytics framework for project management (VIAProMa). This collaborative mixed reality platform allows developers, stakeholders and users to collaborate and administer open-source projects. With the subsequent evaluation of the software artifact, this paper contributes to the understanding of the DevOpsUseXR model [12].

The remainder of the paper is structured as follows. In Sect. 2, we present related approaches in the research. Section 3 describes the data visualizations that we envisioned and realized to support the project management process. After that, Sect. 5 describes further features and steps in the realization of VIAProMa like the collaborative environment. In Sect. 6, we present how we conducted the technical and user evaluations with its results. Section 7 contains a discussion how the realized framework can support project management workflows and which challenges remain for the future. The paper closes with a conclusion and outlook on future work on VIAProMa in Sect. 8.

2 Related Work

The research project lies at the intersection of the fields of mixed reality, immersive analytics and project management. In its original definition, mixed reality describes a spectrum between the real world and a completely computer generated environment [13]. Between these two extreme points, two intermediate forms can be distinguished. Augmented reality (AR) describes a mixture where one can predominately see the real world but there are some virtual elements integrated into the view. Augmented virtuality (AV) is the opposite of this as the user perceives a virtual environment into which elements of the real world are integrated. Since this definition, the term mixed reality has also been applied in slightly different contexts [18]. For instance, mixed reality can also be seen as a form of advanced augmented reality. Whereas AR is only able to overlay virtual content over the real world, mixed reality applications have an understanding of the real 3D space [18]. Therefore, they can integrate 3D objects seamlessly into the real world, e.g. by placing a virtual cup on a real table.

In the field of DevOps, different extensions to the original concept have been formulated. DevOps aims at closing the gap between developers and system maintainers and their conflicting interests [1]. Developers advance the software and would like to deploy updates with new features and bug fixes frequently to clients. In contrast to this, system maintainers try to avoid downtime of the system. Hence, once the software infrastructure is functioning, they aim at altering it as seldom as possible. To resolve this conflict, DevOps introduces a seamless interaction cycle between developers and system maintainers. It contains the phases of development, testing, deployment, monitoring and feedback. Automated pipelines like continuous integration and continuous deployment can support the DevOps cycle. They automatically test the code base and compile the code regularly, thereby catching potential errors in the code early on.

With DevOpsUse, as an extension to DevOps, the participation of the end user in the process was suggested [10]. In each phase, the end user can also interact with developers and maintainers. For instance, during development, a co-design process can happen. The end users can test prototypes and they can give feedback in-between cycles by stating new ideas and requirements. A possible Web tool for social requirements engineering is the Requirements Baazar [16]. Here, communities consisting of developers, stakeholders and end users can gather to formulate, discuss and vote on requirements of projects.

DevOpsUseXR is another expansion of this concept where all these activities can happen in a collaborative mixed reality environment [12]. End users, developers and maintainers can meet in mixed reality to collect ideas and discuss requirements. Users can directly provide feedback about their experience in the mixed reality environment without switching the device. Hence, DevOpsUseXR reduces technological barriers that are imposed on workflows where users would e.g. have to switch between desktop PCs and head-mounted displays.

In our implementation we also make use of gamification. In this concept, game elements are integrated into non-game scenarios to increase the motivation [5]. For instance, it has a positive effect to recognize and visualize the own progress in an otherwise large and complex project. Moreover, story and meaning can be added to activities to increase the perceived importance of the given task and to continuously work on it [17].

Previous research for combining agile development with mixed reality has mainly been concerned with teaching purposes. Applications can be used to simulate projects where Scrum can be applied and to give students practical experiences in an artificial and controlled environment. For instance, Caserman and Göbel created a serious game for teaching Scrum where students get to know Scrum roles, artifacts and the different meetings [3]. Here, other team members and roles are simulated by non-player characters. Radhakrishnan and Koumaditis present a similar approach where an immersive virtual reality environment enables students to learn about agile methodology [15]. Here, users collaborate in the virtual space and have a virtual Kanban board. The team gets a fixed project goal to furnish the interior of a miniature space using agile project management.

Our approach is to use real project data and fetch them from existing project management tools like the GitHub issue tracker. This way, users can inspect real-life data and they can continue working with their favorite productivity tools. Mixed reality becomes another available option to streamline meetings and support the community-building. Within the environment, users can monitor and steer the project.

3 Project Management Data Visualizations

In order to explore the characteristics of DevOpsUseXR, we developed a Visual Analytics framework for Project Management (VIAProMa). It offers a persistent mixed reality environment where developers, end users and stakeholders can collaborate to organize projects in an agile way. VIAProMa is a mixed reality framework for project management. It is based on the Unity 3D engine and the Mixed Reality Toolkit by Microsoft. The resulting application provides cross-platform access as it targets both AR and VR head-mounted displays like the Microsoft HoloLens and HTC VIVE. Moreover, it supports an AR mode for smartphones and tablets based on ARCore. VIAProMa integrates the Photon engine to create a shared experience. With its help, users can collaborate remotely in a shared 3D environment. Its code is available as an open-source repository on GitHub[1].

3.1 Task Cards

In order to provide insights about projects, VIAProMa imports data from the Requirements Bazaar and GitHub. The foundational data object are task cards. They are filled either with requirement items from Requirements Bazaar or issues from GitHub. Users can import new cards using a shelf where they are stored as virtual cards on its boards as depicted in Fig. 1. This shelf offers the possibility to list and filter the content of a selected project. A text input field at the top allows searching for specific cards within the project. Then, the user can drag out individual task cards and place them freely in the 3D scene. This gives a

[1] https://github.com/rwth-acis/VIAProMa.

team manifold opportunities to structure information in the three-dimensional space so that they can be accessed and viewed by everyone and at any time. The space is persistent and therefore, users can access the shared environment independently or synchronously. Since everyone is working in the same environment, updates are immediately propagated to all users and so newest information about the project is broadcast to all.

Fig. 1. Shelf with task cards

3.2 Kanban Boards

Apart from the free-space alignments, users can also construct Kanban boards and add the task cards to them. This tool provides single columns of the board so that users can build individual variations of Kanban boards as depicted in Fig. 2. For instance, instead of the default three categories "to do" "in progress" and "done", other categories can be chosen or additional ones added like "in review". Each column is renamable and its color can be altered in the configurations of the application. Moreover, the column provides scaling handles at its edges so that the area of the column can be scaled. The displayed list of cards in one column will adapt accordingly to the available space by switching into a pagination mode if there are too many task cards to show them all at once. There are no restrictions regarding the alignment of Kanban board columns. Therefore, users can also build three-dimensional variants that e.g. are curved. The "time travel Kanban board" is one variant that utilizes the three dimensional space and can be created with the tool. Here, the team can duplicate the Kanban board each week and only work on the newest copy. Individual copies are placed behind each other. This way, a stack of time slices is formed and the team members can walk past the boards and inspect how they changed over the course of the project.

Fig. 2. Immersive 3D Kanban board

3.3 Contribution and Competence Overview

Another 3D visualization for projects is the contribution overview. The visualization starts out as an empty wireframe box that can be pulled into the scene. After that, users can select task cards from the scene or directly inside of the shelf and mark them as relevant data for the visualization. Internally, the visualization analyzes the given task cards and assigns scores to contributors. For instance, five points are awarded for realizing a task, four points for opening a new one and one point for commenting or voting for a task. The resulting scores are normalized and shown to the user by displaying the name of the contributor and profile pictures with an according scale. Hence, the user will see a circle of arranged profile pictures and names where team members who contributed most to the selected set of tasks are the biggest item of the visualization. The third dimension is utilized by displaying a bar that starts at the back of the profile picture. Its length again signifies the weighted amount of contributions. However, it is also divided into more detailed sections that break up how the score was calculated as shown in Fig. 3.

By looking at these bars, the viewer can identify why somebody is considered a major contributor for the given set of tasks and whether they contributed by implementing code or by engaging in the community around the project. Apart from the contribution overview, the same visualization can also be used for a slightly different purpose. In its settings menu, there is a filter option where the user can type in keywords that must appear on the task cards, e.g. in the title or description. By adding a large amount of cards and filtering for specific topics, the visualization can help identify competences. For instance, if a developer is working on the existing login module of an application, it is possible to add all task cards of the project to the visualization and to search for keywords like “OAuth” or “OpenID Connect” in order to find an expert in this field. The resulting visualization will only show team members who contributed something to task cards that are related to the given keywords. This is a useful tool for the assignment of tasks in large, anonymous open-source projects as developers do not need to know each other but can still make educated decisions regarding the assignment of tasks within the community. Moreover, it can help new developers to get to grips with individual features of the existing application as they can search for the main contributors who previously worked on this feature and can ask them for advice.

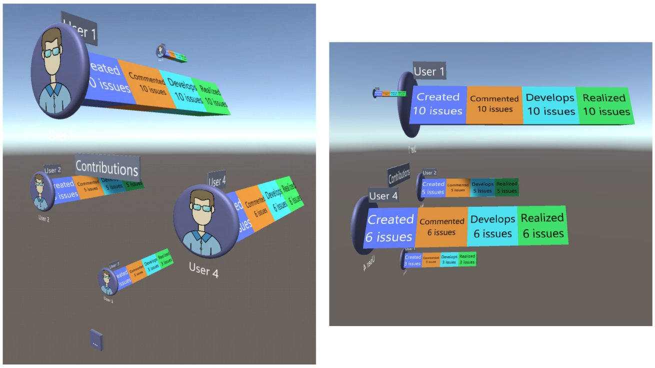

Fig. 3. Competence overview visualization

3.4 Gamified and Non-gamified Progress Bars

In long-term projects it is necessary to monitor the overall progress. VIAProMa supports this with progress bars that can be placed freely in the 3D space. Similar to all previous visualizations, users can instantiate a progress bar in the scene and fill it with task cards. The progress bar will then visualize the ratio of open tasks to tasks that are in progress and tasks that are closed. The progress bar is filled with a green bar for closed tasks and a yellow bar for tasks that are currently in development. The open tasks are represented by an empty section in the glass tube of the progress bar as visualized in Fig. 4. The status of a card is determined implicitly. A task card is open if it is not closed and no developer was assigned to it. The "in progress" label is added if an open task has an assigned developer. The system considers a task as achieved if the corresponding task item is closed. One can add all task cards to track the entire project progress or a sub-set of cards to monitor a single sprint.

In addition to these progress bars, VIAProMa also contains a gamified variant to indicate progress. This separate visualization makes use of a construction metaphor to represent work in the project. Instead of a progress bar, the visualization consists of a skyscraper. The construction of the skyscraper only commences if more assigned task items are realized. Here, finished tasks are represented by revealing the building from the bottom upwards. Tasks that are currently in progress are indicated with a scaffolding as shown in Fig. 5. Open tasks are symbolized by the air above the construction site. Similar to the bars in the progress bar, the height of each of these three sections is determined by

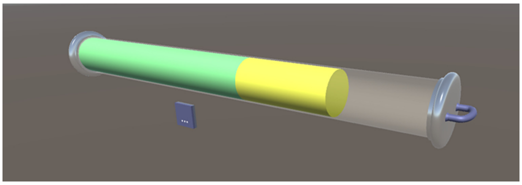

Fig. 4. Progress bar

the percentages of tasks for each category. If one of the buildings contains a lot of tasks that are currently in progress, a small animated crane is also instantiated at the top of the scaffolding. It can act as an indicator for parts of the project that are currently heavily under development. The motivational effect of this progress bar as a gamification element is supported in multiple ways. The application selects the visual style of the building at random from a pool of pre-modelled skyscrapers. Team members can only find out which building is hiding in the visualization by finishing the associated tasks. Moreover, in a project with multiple milestones or sprints, different building progress bars can be created. Over time this forms a city, giving team members the opportunity to visually reflect on the accomplished work and giving them the motivation to further grow the city.

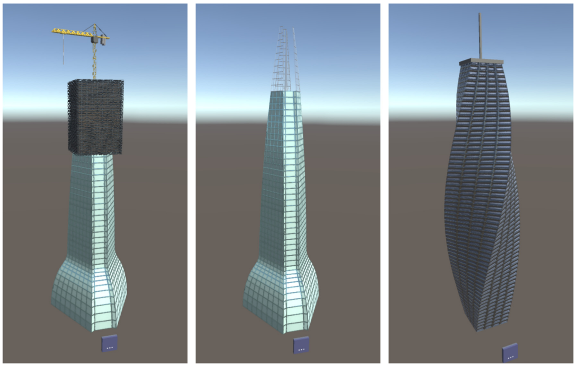

Fig. 5. Different examples of a gamified progress bar

3.5 Commit Statistics

VIAProMa also contains a visualization that does not work with the task data but instead reads information that is provided by the source services of GitHub. In a three-dimensional bar chart, the commit characteristics regarding the aggregated amount of commits per day and hour can be visualized. This 3D diagram helps users to understand the behavior of the developers and to identify peak productivity hours. This is a measure that can help in scheduling meetings. Moreover, the diagram also contains important indication points for stakeholders if the workload is too high or deadlines are set too tight. In this case, the diagram can indicate if more work happened after hours and on weekends. Based on these findings, stakeholders can take according measures by prioritizing and reevaluating the task backlog (Fig. 6).

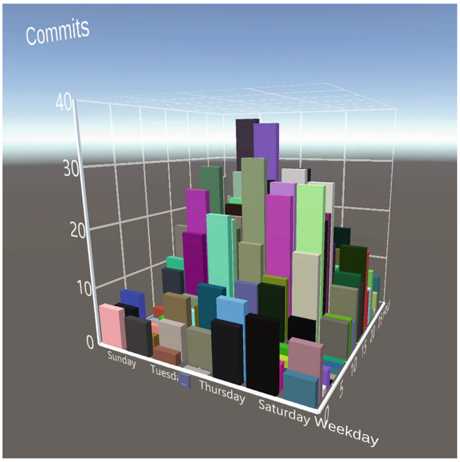

Fig. 6. The commit statistics visualization as a bar chart

4 Use Case Scenarios for VIAProMa

VIAProMa's main intended use case is to analyze and visualize project management data in mixed reality. This functionality can be applied for agile development and it acts as a tool for the DevOpsUseXR concept.

With its synchronous, remote collaboration environment, VIAProMa can especially improve remote conductance of project meetings in the agile workflow. For instance, VIAProMa offers support for the sprint kickoff meeting that is practiced in Scrum. Here, participants can meet in the virtual shared environment, both with AR and VR devices. Since they can see the avatars of others and they can walk around in the virtual space, the experience in VIAProMa imitates a physical meeting. The Scrum master can pull task cards from the shelf in order to discuss them with the developer team and the end users. During this discussion, participants have a three-dimensional space available in order to sort the open task tickets and categorize them spatially. For instance, they can assign meanings to positions in the room, e.g. by collecting tasks that should be tackled in the next sprint in one corner of the room. In contrast to existing 2D desktop-based applications, this also allows for distinguished categorizations. Since they have a 3D space available, teams can establish their own visual language and a sorting scheme that fits their project best. As an example, developers can group cards for the next sprint by clustering them. In the spatial alignment of the card, developers can express the priority of a card by its height in the room. This way, it immediately becomes evident if the next sprint will e.g. focus on new features, designs or bug fixing and the central tasks can immediately be identified. New features can efficiently be assigned to experienced and suitable developers using the competence overview.

Apart from the sprint planning, VIAProMa also offers support for the daily standup meeting that is conducted in Scrum. Within the collaborative environment, VIAProMa breaks up the static spatial nature of a meeting and replaces it with a visual poster session. Similar to a three-minute-madness on conferences, developers can prepare their report in a designated part of the room space by preparing issue cards and visualizations. The daily standup meeting is then conducted by walking through the space and listening to the brief presentations of each developer. Progress bars can be placed persistently in the room and allow stakeholders and developers to continuously track the sprint content. The visualization allows them to notice early on if the progress in a sprint is slower than expected, so that according measures can be taken.

A retrospective meeting of a sprint in Scrum can also be conducted using the virtual space of VIAProMa. Here, developers can instantiate suitable visualizations that show the process in the past sprint. Progress visualizations give an overview whether the sprint has been fulfilled or unclosed tasks need to be moved to a new sprint. A more detailed overview of this can also be set up on the Kanban board. Moreover, the team can identify the major contributors for the sprint or for individual aspects of the sprint using the competence overview.

5 Selected Implementation Details of VIAProMa

Apart from the visualization, we implemented specific features to realize the collaborative mixed reality environment.

5.1 Main Menu

The design of the UI is based on our previous mixed reality project GaMR from which we adapted individual UI elements like the created buttons [11]. One identified challenge based on our experiences with mixed reality development is the placement of the main menu. It must always be reachable but at the same time it should neither clutter the space nor obstruct the field of view. Therefore, we came up with a collapsible main menu. In its collapsed state, it is a small cube. If the user presses the button on the cube, it can unfold and presents its contents. Once a user has selected an option in the main menu, it automatically closes again. This way, the menu is only expanded if the user's attention is actually on the menu and an item is requested in the menu (Fig. 7).

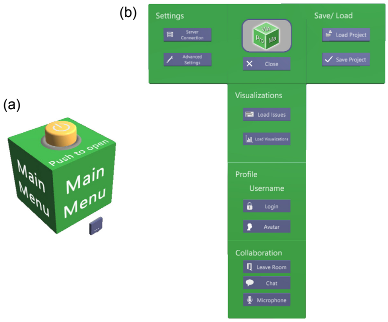

Fig. 7. Main menu in its compact form and the expanded form

5.2 Avatar-Based Collaborative Environment

Since VIAProMa supports remote collaboration, it visualizes the participants in a shared virtual environment as avatars. The avatar consists of a head and a torso. The position and orientation of the user's headset drive the avatar's pose. It is set up in a way that the head will always mimic the real head rotation of the user. The rest of the avatar will follow along with the given movements. Users are able to customize the appearance of the avatar in a dedicated UI menu as illustrated in Fig. 8. Here, a base face, the hair style, hair color, eye color, skin color, glasses and clothes can be chosen. Each avatar wears an ID card which displays the name and their role in the project for improved recognizability.

The collaborative environment is realized using the Photon PUN2 engine. Multiple sessions can be held in parallel as they are separated into virtual rooms. When a user starts the application, the networking module places them in a lobby. Here, they can join existing rooms or create new ones. Once participants are in a room, the collaborative features are enabled. They can see each other's avatars, objects are synchronized, a text chat becomes available and users can set up a voice chat.

Fig. 8. UI for configuring the avatar

5.3 Login System

To identify the user and grant access rights, e.g. on GitHub, we implemented an OpenID Connect login system. The implementation is based on our previous solution deployed in the GaMR project [7]. However, it had to be adapted because Unity's compilation method for UWP switched from a C# project to IL2CPP which first converts the project to C++ source code. During the update process, we also expanded the login compatibility and functionality. Now, the login works both for installed apps on the HoloLens, on Android, for the standalone executable and in the Unity editor. A login button in the application opens the system's default Web browser with the login page of the OpenID Connect provider. We support both GitHub and Learning Layers login as the two login options for the GitHub issue system and the Requirements Bazaar. Once the user has logged in, a redirect takes place that brings the user back into the app and allows the system to read an authorization code. The authorization code is subsequently traded for an access token.

Depending on the platform two different solutions for capturing the redirection are deployed: For native applications on the HoloLens and Android, a custom URL scheme is registered as a deep link for the app, e.g. "viaproma:/". After the successful login, the redirection will lead to this custom URL scheme. Since the application is registered to handle the scheme, it is brought back into focus and an event is triggered which contains all parameters of the redirection. For the platform versions on the PC for the HTC VIVE and the debugging version for the Unity editor, the application opens a temporary server on the loopback address with a free port, e.g. "http://127.0.0.1:12345". The redirection is set up to lead to this given address and therefore, the server notices the request and can extract the necessary parameters of the redirection. The resulting access token is used to get user information like their name to show it on their avatar ID card and to gain write access on project management tools.

5.4 Integration into Existing Productive Processes

A major consideration in the development process of VIAProMa is the maintenance of high levels of efficiency and productivity. Hence, VIAProMa is designed so that it does not disrupt collaborative workflows with accustomed tools like GitHub. Instead, VIAProMa is compatible to GitHub issues and requirements on Requirements Bazaar. Developers can continue using their established workflows by e.g. organizing issues on the Web interface but VIAProMa offers them additional options for remote meetings and analytics insights. It is designed as an additional tool which can be used to streamline synchronous and asynchronous remote collaboration and to gain insights about the project.

VIAProMa imports requirements from Requirements Bazaar and issues from GitHub and abstracts them to general tasks. This is done by a separated frontend-backend architecture. The frontend asks for the task data at the RESTful API of the backend. After that, the backend then fetches the data from the corresponding online API. Before returning them to the frontend, the service-specific data structures are mapped to a generalized task object. This uses the mapping schema shown in Fig. 9. The task object is constructed by considering common properties of tasks, e.g. a title, a description, an assigned developer and a flag whether the task is fulfilled. Hence, the frontend can work with the abstracted tasks and does not need to introduce a new data structure for each data source. For instance, the visualizations on the frontend all work with the general task object and are therefore independent of the data source. Adding a new data source like Gitlab to the system only requires the implementation of the API client on the backend and the mapping of the data structure to the general task object. On the frontend side only the UI needs to be adapted so that the additional data can be requested by the user.

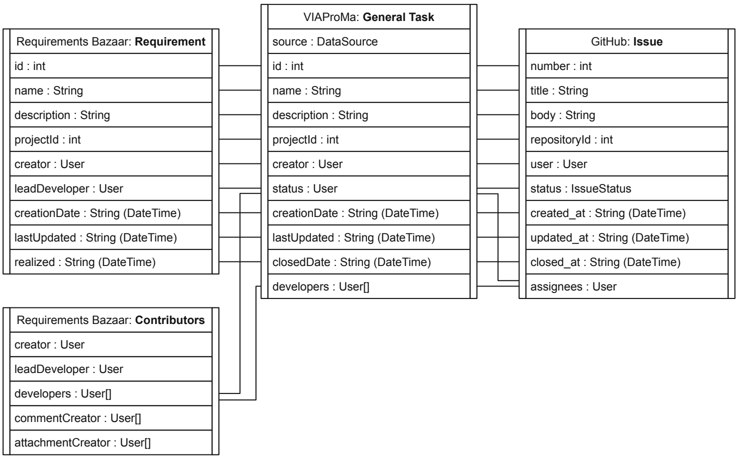

Fig. 9. Mapping of data structures to a general task object

5.5 Persistency

Created 3D rooms are saved by the system so that users can continue working in a scene at a later point. The application automatically saves the scene every five minutes and gives users the option to manually save it. Once a save process is initiated, the scene is converted to a JSON representation. This is executed by a central save-load manager. It keeps track of all savable objects which have a serializer component attached to it. The save call is propagated to the serializer component which is attached to a Unity GameObject. It is composed of multiple serializable classes that determine how the object is saved. This architecture is visualized in Fig. 10. Each serializable component implements an interface to execute the save and load calls. This way, reusable serializable modules emerge. For instance, there can be a module for saving the position, rotation and scale of an object. This component can then be reused on all objects for which this property should be stored. With a save call, the serializable components construct a key-value dictionary for each object. The serializer collects these dictionaries and merges them together into one key-value store for the object. This key-value store is also assigned an ID so that the object can be recognized later on. Every serializer returns its constructed key-value store to the save-load manager which converts it to a JSON string. The JSON string is then sent to the backend where it is stored.

For loading, this process is reversed. The JSON string of the given session is retrieved from the backend server and parsed into the key-value stores. After that, the save-load manager compares if an object with the given ID already exists in the current session. If this is the case, its values are overwritten and otherwise, a new object is created. The values are applied again by the serializable components which define how to read the keys and values from the dictionary.

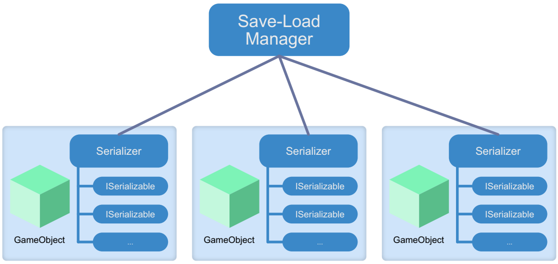

Fig. 10. Architecture of the save load system

6 Evaluation

We conducted a user evaluation and a technical evaluation to gain insights about the usability of the framework and its usefulness for agile project management in mixed reality.

6.1 Evaluation Setup

A user evaluation was conducted with the goal to assess the usefulness of the 3D visualizations, the usability of the application and the utility of the collaborative environment. Thus, fifteen computer science students worked with VIAProMa at our lab. Its setup follows the insights and experiences described by Dünser and Billinghurst [6]. As the first step, participants were asked to fill out a pre-questionnaire which collects data about their demographics and asks about previous experiences with mixed reality, as well as agile development practices. After filling out the form, they were assigned a Microsoft HoloLens or an HTC VIVE Pro and started the application. The participants worked with data from real open-source projects on Requirements Bazaar and GitHub. We provided milestones for these open-source projects and asked the participants to find a suitable spatial ordering to represent the milestone. Additionally, the participants should make use of the visualizations and Kanban boards in order to express the progress of the milestone and to gain insights on the project status. They were also asked to identify a suitable developer for a given task based on the competence visualization. While the participants were interacting with the application, we observed their behavior using the device portal of the Microsoft HoloLens and the casted preview feed of the HTC VIVE Pro. Once participants established a spatial structure, we joined the collaborative virtual environment so that the user could present the gained insights about the project. Finally, participants filled out a post-evaluation questionnaire. This form mainly contained qualitative questions on a Likert scale to answer the initial research questions, followed by a free-text section where participants could give more general comments and feedback.

6.2 Evaluation Results

In the pre-questionnaire, it became evident that nobody has had previous experiences with the Microsoft HoloLens. However, with the HTC VIVE, it was different as 47% reported that they had already used it. Here, five participants had used it one to three times prior and two participants indicated more than eight sessions with the HTC VIVE. 93% of participants did not develop 3D applications with the Unity engine. 87% of the students were familiar with concepts of agile project management. In detail, 11 participants named Scrum as their most known agile methodology, whereas Kanban and extreme programming were reported by two participants each. One person mentioned the scaled agile framework (SAFe).

The results of the final questionnaire show that the participants thought that the application gives an insight about the status of a project. Here they answered in the range from three to five on the Likert scale where 5 indicates strong agreement. The median was located at five and the lower quartile at 4.5. Regarding the assignment of a task to a competent developer, participants agreed with a median scale of four and a deviation between three and five that the framework helped them. For the visualizations, the issue cards were deemed the most useful as shown in Fig. 11. All other visualizations go a median score of four with a minimum score of three and a maximum score of five. Concerning the arrangement of the menus, the users stated at a median of four and an upper quartile of five that the menus are tangible and well-structured. The possibility to customize the Kanban board was well-received a median of four and a lower quartile of four, as well as an upper quartile of 4.5. All participants agreed with a score between four and five that the charts helped them understanding the status of the project. Regarding the usability of the application, users on median assessed the intuitiveness with a score of five and also with an upper and lower quartile of five. Even though the participants had little prior experience with mixed reality technologies, they considered the positioning of the 3D models in the space with a median of four. Here, the lower quartile resulted in a score of four and an upper quartile of five. We also asked them whether the application improves remote collaboration. The answers lied between a score of three and five with a median of five and a lower quartile of 4.5. The participants regarded the customization process of their avatar as fun with a median score of 5. Finally, 73% of the participants made use of the whole 3D space when arranging the imported objects like the task cards and visualizations to structure the project.

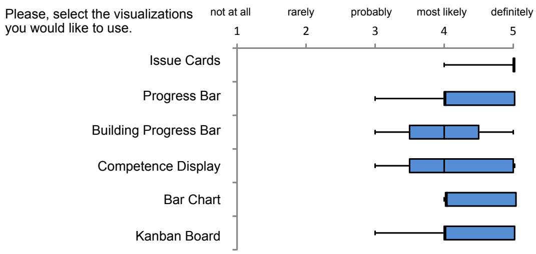

Fig. 11. Results regarding the usefulness of the visualizations

6.3 Technical Evaluation

We monitored the application's performance using the mixed reality toolkit's profiler window, the device portal of the Microsoft HoloLens and Unity's profiler. The framerate and memory consumption were evaluated separately for each supported platform. On the Microsoft HoloLens, a stable 60 frames per second are reached. This is the recommended value by Microsoft[2]. On a PC with a NVIDIA GTX 1070ti graphics card and an Intel i7 processor, we reached 90 frames per second for the VR device. On Android, ARCore caps the framerate at 30 frames per second to reduce the battery drainage. With this value, movement is still perceived as smooth and the framerate requirements are not as high as the screens are not directly in front of the eyes of the user.

[2] https://docs.microsoft.com/en-us/windows/mixed-reality/develop/platform-capabilities-and-apis/hologram-stability#frame-rate.

7 Discussion

The main challenge in the design of VIAProMa was to offer compatibility and efficiency for the usage in the day-to-day business of the users. We are aware that mixed reality cannot replace existing agile project management tools around which established workflows have already been constructed. Nevertheless, it can add an additional layer of interaction and insights. It has the potential to improve meetings and give an overview of the project status. The evaluation results show that its value lies in the fact that remote meetings get streamlined and that the teams can immerse in a three dimensional space which can be shaped to their liking. Moreover, it adds additional information that is not directly incorporated in existing project management tools, e.g. the competence overview. We plan to conduct an extended evaluation on a larger scale to gain meaningful results with newly added features.

The developed framework VIAProMa supports the feedback phase of the DevOpsUse cycle. It provides a shared mixed reality environment for remote collaboration where the different stakeholders, developers and end users can meet in a persistent environment. Users evaluated the 3D interface as intuitive. It allows end users without development knowledge to participate and understand the project progress. In VIAProMa, they can directly interact with tasks that are represented as cards and with visualizations as opposed to more technical tools that are fitted for developers. It supports both requirements engineering and sprint planning.

8 Conclusion and Future Work

In this paper we presented the VIAProMa framework which combines mixed reality with agile project management. We created and implemented a series of visualizations that allow users to gain insights about a given open-source project. The visualizations are based on task data by the Requirements Bazaar and GitHub. Accordingly, we adapted the UI to the mixed reality environment to ensure a good usability. Since VIAProMa should support meetings in the agile workflow and provide a persistent environment for this, collaboration features were added. Objects and visualizations can be shared with participants in real-time and remote contribution is made possible by representing users as customizable avatars. We evaluated the resulting implementation in a technical and a user evaluation. The results show that VIAProMa provides a series of use cases to support agile development. The visualizations were well-received by participants and the majority utilized the entire space to organize their project visually. In particular, the issue cards and bar charts with the commit statistics were found interesting by users. VIAProMa supports DevOpsUseXR by enabling end users to participate in developer meetings and to co-design the final product.

A series of solutions emerged from VIAProMa which fulfill common requirements for mixed reality projects. For instance, mixed reality applications in the realm of learning need to identify the current user with a login system to attribute learning success. Here, our cross-platform capable OpenID login solution can fulfill this requirement. Hence, we are currently extracting and modularizing the realized solutions. Like the VIAProMa framework itself, the resulting software artifacts will be made available publicly under an open-source license. This way, developers can directly import configurable modules to use them in their own mixed reality projects. Future work will also focus on extending the DevOpsUseXR support. For instance, VIAProMa could be extended by co-design tools that allow users to create mockups for UI elements. Another possible extension would be the integration of usage analytics in the VIAProMa environment. This way, collaborators could not just analyze the development process but also the way how users interact with their final application. Because of the data abstraction layer, VIAProMa can be extended with new data sources like Gitlab or Trello. This way, the platform can be compatible to a wider range of management tools. This will allow projects to integrate VIAProMa into their workflow while retaining their preferred development tools. The existing visualizations could also be used to inspect learning analytics data of individual students but also of entire courses. So, VIAProMa could also be transformed into a personalized student-mentor environment as a scalable solution for individual feedback.

After the completion of this start phase, the collaborative platform was operational, the connection to the project management tools functioned and their data could be visualized in mixed reality. An item of future work that is already in progress concerns using VIAProMa in teaching. We will continue to use VIAProMa as the foundation for software projects in our mixed reality labs that teach agile mixed reality development. An instance of the lab consists of short-term projects of different student teams who implement additional features in VIAProMa to create useful collaborative tools.

All in all, VIAProMa proves to be a suitable framework for agile project management and DevOpsUseXR. In its creation, a stable shared open-source platform emerged that can function as the basis of further collaborative mixed reality projects for cross-platform interactions.

References

1. Bass, L., Weber, I., Zhu, L.: DevOps: A Software Architect's Perspective. Addison-Wesley Professional (2015)
2. Billinghurst, M., Kato, H.: Collaborative augmented reality. Commun. ACM **45**(7) (2002). https://doi.org/10.1145/514236.514265
3. Caserman, P., Göbel, S.: Become a scrum master: immersive virtual reality training to learn scrum framework. In: Ma, M., Fletcher, B., Göbel, S., Baalsrud Hauge, J., Marsh, T. (eds.) JCSG 2020. LNCS, vol. 12434, pp. 34–48. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-61814-8_3
4. Dabbish, L., Stuart, C., Tsay, J., Herbsleb, J.: Social coding in GitHub: transparency and collaboration in an open software repository. In: Poltrock, S. (ed.) Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work, p. 1277. ACM, New York (2012). https://doi.org/10.1145/2145204.2145396

5. Deterding, S., Dixon, D., Khaled, R., Nacke, L.: From game design elements to gamefulness: defining "Gamification". In: Lugmayr, A., Franssila, H., Safran, C., Hammouda, I. (eds.) The 15th International Academic MindTrek Conference, p. 9 (2011). https://doi.org/10.1145/2181037.2181040
6. Dünser, A., Billinghurst, M.: Evaluating augmented reality systems. In: Furht (Hg.) 2011 - Handbook of Augmented Reality, vol. 47, pp. 289–307. https://doi.org/10.1007/978-1-4614-0064-6_13
7. Hensen, B., Koren, I., Klamma, R., Herrler, A.: An augmented reality framework for gamified learning. In: Hancke, G., Spaniol, M., Osathanunkul, K., Unankard, S., Klamma, R. (eds.) ICWL 2018. LNCS, vol. 11007, pp. 67–76. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-96565-9_7
8. Hoda, R., Salleh, N., Grundy, J.: The rise and evolution of agile software development. IEEE Software **35**(5), 58–63 (2018). https://doi.org/10.1109/MS.2018.290111318
9. Klamma, R., Ali, R., Koren, I.: Immersive community analytics for wearable enhanced learning. In: Zaphiris, P., Ioannou, A. (eds.) HCII 2019. LNCS, vol. 11591, pp. 162–174. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21817-1_13
10. Klamma, R., et al.: DevOpsUse - Scaling Continuous Innovation: D6.3: Learning Layers Project Deliverable (2018)
11. Koren, I., Hensen, B., Klamma, R.: Co-design of gamified mixed reality applications. In: Proceedings of the IEEE ISMAR 2018 Workshop on Creativity in Design with & for Mixed Reality, pp. 315–317 (2018). https://doi.org/10.1109/ISMAR-Adjunct.2018.00094
12. Lefrere, P., et al.: Roadmap and Sustainability: WP 8 — D8.4. http://wekit.eu/wp-content/uploads/2019/04/WEKIT_D8.4.pdf
13. Milgram, P., Kishino, F.: A taxonomy of mixed reality visual displays. IEICE Trans. Inf. Syst. **E77-D**(12), 1321–1329 (1994)
14. Piumsomboon, T., et al.: Mini-Me. In: Mandryk, R., Hancock, M. (eds.) Engage with CHI, pp. 1–13. The Association for Computing Machinery, New York (2018). https://doi.org/10.1145/3173574.3173620
15. Radhakrishnan, U., Koumaditis, K.: Teaching scrum with a virtual sprint simulation: initial design and considerations. In: Teather, R.J., et al. (eds.) 26th ACM Symposium on Virtual Reality Software and Technology, pp. 1–2. ACM, New York (2020). https://doi.org/10.1145/3385956.3422107
16. Renzel, D., Behrendt, M., Klamma, R., Jarke, M.: Requirements Bazaar: Social Requirements Engineering for Community-Driven Innovation. In: 21st IEEE International Requirements Engineering Conference: RE 2013, Piscataway, NJ, pp. 326–327. IEEE (2013). https://doi.org/10.1109/RE.2013.6636738
17. Sailer, M., Hense, J.U., Mayr, S.K., Mandl, H.: How gamification motivates: an experimental study of the effects of specific game design elements on psychological need satisfaction. Comput. Hum. Behav. **69**, 371–380 (2017). https://doi.org/10.1016/j.chb.2016.12.033
18. Speicher, M., Hall, B.D., Nebeling, M.: What is mixed reality? In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. CHI 2019, pp. 1–15. ACM, New York (2019). https://doi.org/10.1145/3290605.3300767
19. Tam, C., Da Moura, E.J.C., Oliveira, T., Varajão, J.: The factors influencing the success of on-going agile software development projects. Int. J. Project Manage. **38**(3), 165–176 (2020). https://doi.org/10.1016/j.ijproman.2020.02.001