GRANT journal
ISSN 1805-062X, 1805-0638 (online)
Vol. 12, issue 01

EUROPEAN GRANT PROJECTS | RESULTS | RESEARCH & DEVELOPMENT | SCIENCE

Agile management of mobile application development for the library of the scrum framework

Rebecca Frančáková[1]
Patrik Flegner[2]
Ján Kačur[3]

[1] Institute of Control and Informatization of Production Processes, Faculty BERG, Technical University of Košice, Boženy Němcovej 3, 040 01 Košice, Slovak Republic; rebecca.francakova@tuke.sk;
[2] Institute of Control and Informatization of Production Processes, Faculty BERG, Technical University of Košice, Boženy Němcovej 3, 040 01 Košice, Slovak Republic; patrik.flegner@tuke.sk;
[3] Institute of Control and Informatization of Production Processes, Faculty BERG, Technical University of Košice, Boženy Němcovej 3, 040 01 Košice, Slovak Republic; jan.kacur@tuke.sk;

Grant: VEGA 1/0264/21
Název grantu: Application of modern methods in the analysis and modeling of technological and other processes used in the acquisition and processing of earth resources in order to optimize them.
Oborové zaměření: IN - Informatika

Abstract Currently, more and more companies already have mobile applications or have started working on them or will start developing them. This paper will describe the agile management for the development of a mobile application for the Ján Bocatia Public Library, explain the importance of the SCRUM Master position and analyze the current state of libraries in Slovakia. An online meeting was held with the project sponsor to find out the problem of the library and the target group for which this mobile application will be developed. Finally, the paper offers our proposal for a mobile application to attract new readers and our first experience with this framework used in the project and its impact on the mobile application. Last but not least, suggestions for possible improvements to the application in the future.

Keywords Agile management, agile methodology, app development, Scrum Master

1. INTRODUCTION

Recently, mobile applications are quite popular. They make people's work and life much easier. Instead of lengthy searches for individual websites or organizations, it is necessary to download an application to a smartphone and open it with a single click. Thus, the user always has the applications at hand on the desktop and can easily open them anytime and find what he is looking for. In addition, some applications are available offline, so users can check things even when they are out and get what they need.

A mobile application, or app for short, is a software application created specifically for mobile devices such as smartphones, watches, emulators, tablets, and other mobile devices. A mobile application is the opposite of desktop applications (which run on desktop computers) and web applications (which run in web browsers on a smartphone rather than directly on the smartphone). The first mobile applications were from manufacturers of mobile operating systems that needed their users to create practical methods for checking e-mails, making calendar entries, viewing the stock market, weather reports, and other essential functions. Over time, however, people began to ask more and more for a more diverse range of applications. So programmers began developing mobile games, company applications, production automation, practical applications using GPS, order tracking, and various fun applications for children and adults. Generally, apps are downloaded from app stores, which are a type of digital distribution platform. Each digital distribution platform has its app store [1].

The use of mobile applications is increasingly prevalent among mobile phone users. The researchers found that mobile app usage depends on the user's location and time of day. Applications are generally small, self-contained software units with limited functionality. Mobile apps are increasingly important in healthcare and, if designed correctly, can bring many benefits. Some apps are free, others may be pre-paid or prepaid, and some apps include ads. Revenues for sold apps are usually split between the app creator and the app store. Depending on the mobile platform, the same app may cost a different price.

One of the most widely used operating systems, mainly used in mobile devices, is Google Play. On March 6, 2021, Google Play was launched, bringing together Android Market, Google Music, Google Movies, and Google eBookstore under one brand. Google Play, usually referred to as the Google Play Store, was once referred to as the Android Market. It is an online distribution service that is operated and developed by Google. It currently provides several digital content types, such as games, music, books, movies, and TV programs. In addition, some apps on Google Play are available either for free or for a fee.

The next most used operating system is iOS. First, iPhone OS was developed based on macOS for the original iPhone, and later it was adapted for other mobile devices such as the iPod Touch multimedia player and the iPad tablet. iOS is a mobile operating system created and developed by Apple. The term iOS also included versions running on iPads until the launch of iPadOS in 2019 and iPod Touch

devices. After Android, it is the second most widespread mobile operating system in the world. Major versions of iOS are released periodically every year [2].

1.1 Scrum

Software development teams often use Scrum (for example, a mobile application) [11]. It is a simple, agile framework for effective team collaboration on complex (complex) products. For these reasons, Scrum was implemented in a mobile application development project to modernize library services [12].

Scrum is one of the most popular agile frameworks, which represents a set of principles: deliver a functional product in short cycles, guarantee fast feedback, enable continuous improvement, and facilitate rapid adaptation to environmental influences and changes [13], [14], [15]. It uses agile thinking to develop, deliver and maintain products with an emphasis on software development. The team divides its work into objectives [16]. Goals can be accomplished in time-bound iterations called sprints [05].

Smaller teams communicate better and are more productive than larger teams. If the Scrum team is too large, it should consider reorganizing into multiple cohesive Scrum teams. They should share the same product goal and backlog [17]. The Scrum team is responsible for all activities related to the product, from collaboration with stakeholders, experimentation, research and development, and whatever else is needed [18]. The entire team is responsible for creating a valuable and helpful increment each sprint [03].

The basic unit of Scrum is a small team called a Scrum team. A scrum team is a cohesive unit of professionals focused on one common product goal.
Within Scrum, 6 phases of the software development life cycle are repeated in each sprint (see Figure 1) [04].

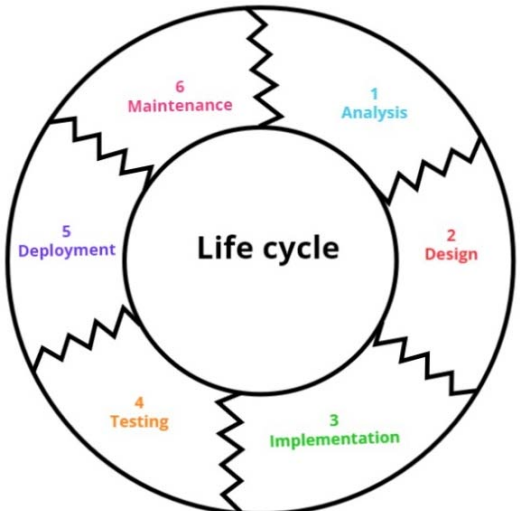

Figure 1 Development life cycle

One of the positions in the Scrum team is the Product Owner, who is responsible for achieving good business results. The Product Owner represents product stakeholders and the voice of the customer during the project. The product owner defines the product in terms of customer-centric outcomes, adds them to the backlog, and prioritizes the backlog based on importance and dependencies [19], [20].

Another position is the Scrum Master, responsible for removing obstacles to the team's ability to meet product goals. A Scrum Master is not a team leader or a project manager. Instead, the Scrum Master represents a barrier between the team and various distractions [06], [07].

The last position in the Scrum team is the developers, who perform the work to create value increments at the end of each sprint. The term developers refer to anyone who plays a role in developing and supporting a system or product. Among developers can be included people who, for example, do analysis, design, development, testing, technical communication, document, and similar [21].

In a Scrum team, ceremonies ensure regularity and minimize meeting needs. Therefore, all ceremonies have a time limit. The basic unit of development in Scrum is sprint. A sprint is a time bound; that is, the length is agreed upon and determined in advance for each sprint. The duration is usually between one week and one month. The sprint length is typically set to two weeks [08].

At the end of each sprint, two events are done [09]: sprint review (progress demonstrated to stakeholders), sprint retrospective (identify lessons and improvements for next sprints within the team), [22], [23].

Another unit in Scum is Sprint Planning. Each sprint starts with sprint planning, the goal of which is to define the tasks for the upcoming sprint to: agree with the Scrum team on the sprint goal, a brief description of what the team will deliver by the end of the sprint, based on the priorities set by the product owner, select the backlog items of products that contribute to the achievement of the goal will create sprint backlog items by discussing and agreeing among themselves which items should be completed during this sprint and which will be completed in the next sprint. The maximum sprint planning time is limited to eight hours for a 4-week sprint [24].

During the sprint, the Scrum team should meet every day, which is usually performed standing up, so that this Daily Scrum does not take unnecessarily long. Daily Scrum belongs to other basic units. The team comes prepared for this meeting. Daily Scrum: is aimed at checking progress towards the sprint goal. It should always take place at the same time and place every day and is time-limited. It should take a maximum of 15 minutes, and the team decides, even if only developers should speak, it does not contain discussions and the like [25].

The penultimate ceremony is the Sprint Review. At the end of each sprint, a Sprint Review should be carried out, where the team: presents the completed work for the given sprint to the stakeholders, asks for feedback on the increment of the finished product, discusses unfinished work, tasks (planned or others), suggestions for future work, improvements (or instructions on what the Scrum team can work on further) are accepted [26].

The last ceremony is the Sprint Retrospective, which is performed at the end of the sprint, after the Sprint Review. Three questions need to be considered in retrospectives:

- *What went well during the sprint?*
- *What did not go as expected?*
- *What could be done differently in the next sprint?*

In Scrum, Scrum artifacts also present work or value, which creates transparency and opportunities for review and adaptation [27]. Scrum artifacts are designed to maximize the transparency of important information and for everyone to understand the artifacts

equally [28]. Scrum artifacts include Product Backlog, Sprint Backlog, and Product Increment. The increment is one of the important artifacts of Scrum. A product increment integrates all completed Product Backlog items during a sprint. Likewise, the product increment integrates all completed product backlog items in a particular sprint. In a project, product increment is the integration of all completed sprint backlogs [10], [29], [30].

1.2 The client of the project

The client of this mobile application development project called Book avatar was the Jan Bocatia Public Library, located in Košice in cooperation with the Košice self-governing region. The public library of Jan Bocatia is the oldest and, at the same time, the largest library in Slovakia. From the name of this public library, it can be deduced who this library was named after. It is clear that it is Jan Bocatius.

The library's brief for the Scrum team, which aims to create a mobile application that will facilitate and help readers find what they are looking for and recommend books they like from the comfort of their homes, was as follows:

- create a mobile application that will popularize book lending,
- find out data about the readers (target group),
- relieve librarians (e.g., from repetitive questions),
- better inform readers (e.g., about news, events, etc.).

With this mobile application, the library wants to make it easier for readers to use the library in a new, modern way and relieve librarians of constantly repeating questions and requests that have already been answered several times.

It was performed research to see if any of the libraries already had a mobile app to get inspiration, see if the library had any competition, and come up with something new to the market that wasn't there yet.

The features that most interested us as readers were:

- A clear list of what the given reader is currently reading.
- Reading prompt. It mainly serves as motivation to read more books.
- Recommendations based on what the reader liked.
- There was found the following attractive features from other apps and sites:
- A reader's card in the application so that the reader does not have to carry it with him all the time or if he forgets it at home.

2. METHODS

The Scrum framework was implemented in the Book Avatar project to create a mobile application for VKJB and to, increase library traffic, and facilitate book search.

First of all, at the beginning of the project, the whole team met with the client (client) of the project directly with the Public Library of Jan Bocatia. The goal of this meeting was to find out and analyze the required functions of a digital product, such as: "*What digital product is going to be created? What is the target group? What functions should a mobile application have?*".

Furthermore, a survey was made in Slovakia regarding libraries to find out if any of the libraries already have a mobile application. If none of the libraries had a mobile application, an application whose main theme is books was sought. Google was used to search the Internet for various applications and sites, the main topic of which is books. The main reason was an inspiration.

Seven team members worked on the Book Avatar project. The team consisted of 1 Scrum Master, 1 Product Owner, 2 Designers, and 3 Developers. When assembling a team, not only is its size important but mainly the necessary experience and knowledge of individual team members. The project was described from the point of view of the Scrum Master - one of the most critical positions in the project. The scrum team worked on this project for five months, from January 2021 to June 2021. During the project, ten sprints took place, with one sprint lasting two weeks.

After the sprint's end, the team's performance was evaluated, such as the number of completed tasks, and uncompleted tasks, what problems arose, and how individual issues were solved. This evaluation was done after each sprint, a total of ten sprints were evaluated.

After the end of the project, a presentation was made for the Jan Bocatio Public Library, for which this mobile application was developed. At the end of the presentation, the library was given a space to comment on the created mobile application.

3. EXPERIMENTAL SETUP AND DATA

As mentioned above, our goal was to create a mobile application by implementing the agile Scrum framework.

Currently, the library is mainly visited by the age group 30 years and above. The younger age group, between 15 and 18 years old, rarely goes to the library. They only borrow books they need for compulsory reading or as study material for a final thesis. This age group prefers mobile phones, which they almost always have somewhere at hand and have various mobile applications installed, where they have all the necessary and unnecessary things. It's easier than manually entering a page several times a day. There is only a need to search for the given application once, install it on our phone, and it is always at hand.

Many of us cannot imagine life without mobile applications. Applications can not only make life easier but also more pleasant. That is why the Public Library of Jan Bocatia in Košice decided to introduce digital products and thereby innovate the field of libraries by creating a mobile application - a book avatar. The book avatar will serve as a personal assistant in the library environment. It will always be at hand, wherever the team members are. It will enable pre-registration directly from the mobile phone, when basic personal data do not have to be entered in the library, but, for example, directly from the comfort of home. Only the reader in question comes to the library to show a valid identity card and sign. In the case of a minor, his legal representative signs. Time can be saved even if the reader only comes to pick up books that are pre-ordered through the given application. If the books have already been borrowed, they can be quickly reserved so no one else can borrow them. After returning the given title to the library, the reader will receive a notification directly from the application about pre-paring the reserved books and their subsequent borrowing. Notifications can be set whether the user wants them turned on or not.

The most crucial function will be book recommendations based on previously borrowed or previously searched books. This feature will make the book search process easier for readers, and librarians will also not be confronted with questions like "*Please advise me of a book; I read this, and I liked it. Is there anything else similar?*". After the initial meeting with the client, a user persona was created.

The client's goals were defined, and a map called the Journey map was created. It's a map of the user and what they're doing. It had to be known for whom and how it was approached.
A competitive analysis was then researched and created to see what the environment the product will be brought into looks like. In our case, an overview of library applications was made and whether any library in Slovakia has a mobile application.

Subsequently, a problem framing was done, with the fact that at the end of it, it was found that there were assumptions about what would probably be created for that client. It is good, mainly because it is possible to communicate with the client more quickly and direct him somewhere when going into a sprint.

During the project, there were several challenges for the team, such as: finding out the data about the target group, market analysis, propose a modern design, verify the functionality of the design, Android operating system, and communication with Tritius.

The Scrum Master coached the team to follow agile practices and agile mobile application development process. Ceremonies were observed - basic Scrum meetings such as Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective. The Scrum Master planned and, at the same time, facilitated (that is, led) these meetings. He encouraged people when they were not speaking to speak, helped with asking questions, and kept time. Once he set a time for the meeting, he had to make sure that everything that was planned could be discussed so that it did not happen that only one of the three topics was discussed in an hour-long meeting. In practice, when the meeting was finally arranged, everyone came, and when the Scrum Master saw that the discussion on a topic was taking longer, he preferred to hold another meeting so that it could be discussed in detail. He did not support the idea that time must be respected, at the expense of quality, that someone would be stopped in the middle of a sentence, and thus nothing would be solved. The goal is not to go through all the questions but not to solve any of them. So he planned it so that it would all be done in that time and that all that needed to be said would be said.

The main task of the Scrum Master was to establish the Scrum process (see Figure 2), which defines roles, artifacts, and ceremonies.

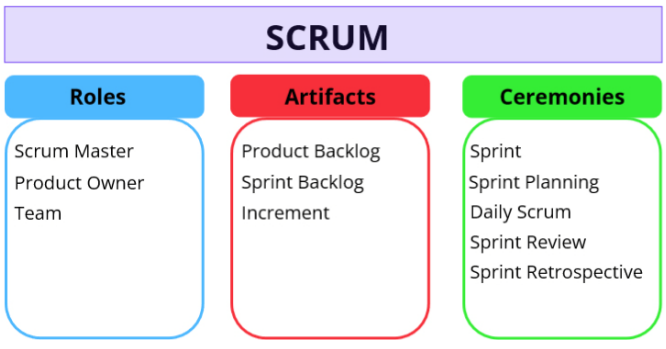

Figure 2 Scrum process

The Scrum process brought roles for which the development team needed to be trained. In this case, the Product Owner was a member of a team that was technically focused and was not management-oriented before this project. The role of the Scrum Master was, therefore, to coach him to understand the basic principles of Scrum and his role as a Product Owner. At the beginning of the project, he formed a bond with the Product Owner when he and the Product Owner had to spend some time together. They arranged several meetings where they first got to know each other and later agreed on other processes they set up. Communication is everything. Thanks to effective communication, they could work with the team to find solutions to problems. He maintained an open-door policy and built trust within the Scrum team and among stakeholders while maintaining a positive attitude—helping the project succeed. For example, the Scrum Master helped the team achieve the project's goals by assisting them with everything they needed to make the work go smoothly, eliminating problems, and motivating the team to achieve better results. It was necessary to motivate the team and protect it from external influences that could reduce it to the set goal. He solved problems of a different nature from the type: the customer has not replied for two days. He tried to find the customer, saying that the team would be late if he didn't say something. He also solved internal problems, unless someone didn't get along with someone or someone had a communication problem with someone, etc. It was important that he supervise the team, that the team observe ceremonies, and that everyone be punctual.

The activities of the Product Owner were not focused on the team as with the Scrum Master but on the product. His main task was taking care of the Product Backlog, where he wrote down the tasks and regularly created and wrote down the requirements that needed to be developed. In addition, he was responsible for the product vision and product development strategy, working with the development team, ensuring the team understood the product requirements correctly and completing the requirements acceptance criteria. During the mobile application development, the Product Owner ensured communication with the customer, sufficiently explained the problems that needed to be solved and communicated all this to the team. He also prioritized the tasks that needed to be done first. He had a technical education, which was not a necessity. Still, it was a great advantage, and he could advise the team on how to move forward in developing the mobile application.

The lists through which basic information was shared with the Scrum team during project management are called Scrum artifacts. There are three main artifacts: Product Backlog, Sprint Backlog, and Product Increment, which will be discussed in the next section.

The first thing to start with was the Product Backlog - it was nothing more than a simple list of all the tasks and product requirements that needed to be fulfilled. This document is never finished, which is why Agile or Scrum differs from Waterfall or classic project management, where a detailed plan is laid out at the very beginning. Here the solution went into the unknown. The library did not yet know what it wanted, and it was unclear what it should do, how, and why it should do it. It wasn't planned all the way. However, an idea (i.e., this task) was developed. Before anything was written in the Backlog, research was done first. What would be done was clear, and the most significant possible competition was found. As much as possible was copied from the competition, and these ideas were further developed. During several sprints, it was possible to brainstorm more than make or try different designs/technologies and pre-sent them to the customer. He chose what he liked more and what he didn't like, and based on that feedback, and the Product Backlog was updated.

The goal of the sprint was also determined. A goal was set because it was a good reminder every time they looked at the Product Backlog of what needed to be accomplished. In addition, it helped the team to focus on the work. Figure 3 shows the entire Product Backlog, divided into three main parts: To do, In progress, and Done.

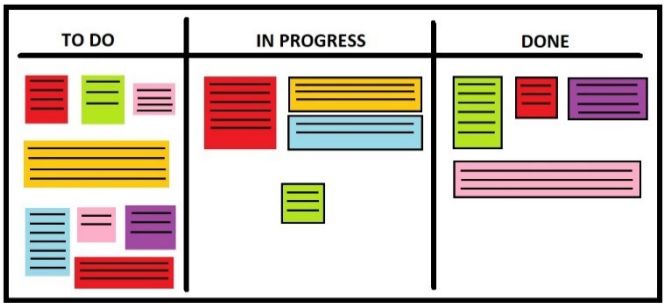

Figure 3 Product Backlog

Whenever the Backlog was viewed, it was possible to see the status of individual tasks, how many tasks still need to be done, how many tasks are still being worked on, and what has already been completed. The Product Backlog was mainly the responsibility of the Product Owner, who was responsible for prioritization; that is, the essential things with the best-detailed descriptions were at the top of the Backlog. It was an organized list of all "User Stories", which included: Login, Registration, Reader Loans, Anonymized Borrowing History, Reservation, Availability of a Work in the Library, Search, Filtering Records, Book Recommendations, and Reviews, i.e., book ratings. Priorities included Login, Reader Loans, Reservation, Search and Recommend books. They changed according to the interests or requirements of the customer. The Product Owner himself did not do it. It was good when the whole team helped in the beginning because they asked various interesting questions that, for example, the Product Owner would not usually have thought of and which the Product Owner then asked the customer.

The meetings were held online using Google Meet. There was no fixed time duration during this meeting, so the length of the meeting was adjusted as needed. It was attended by the entire team, including the Scrum Master, Product Owner, Designers, and Developers. The ten User Stories were divided into 40 tasks, i.e., the items that the Product Owner presented to the team and asked which, according to the group, are the most important or can be started to work on. As if it has been combed and the individual tasks have been put in order. It is how the team knew how to help the Product Owner. The Product Backlog was created for two sprints ahead. It was important for the team not to stand still and do something.

Sprint Backlog was the result of Sprint Planning - which was almost the same as Product Backlog, only for that one sprint. So that was the list of User stories done in the next sprint. In the first sprint, three items were selected for the Sprint Backlog. The team did not meet these items. In the second sprint, two items were selected. The team also did not complete these items, so they were moved to the next sprint. In the third sprint, two items were selected, and five pre-delivered unfinished items were added. Here the team managed to complete three items successfully. Incomplete items remained four. Six items were selected in the fourth sprint. The team finished all six items out of six items.

Eight items were selected in the fifth sprint. The team managed to complete five items. The team did not manage to finish 3. Therefore, they were moved to the next, 6th sprint. In the sixth sprint, six items were selected, and the previous three unfinished. In this sprint, the team managed to complete eight items. Two items and the previous unfinished item were selected in the seventh sprint. It fulfilled two items. In the eighth sprint, three items were selected, and the previous one was unfinished. The team completed all four items. Seven items were selected in the ninth sprint. This successfully completed all seven items. In the last tenth sprint, the last four items were selected. The team met all of these items.

The Product Increment was the design proposal in the first two sprints, but nothing from the Product Backlog. In the third sprint, the team managed to make the basic layout of the application, which was the first Product Increment. In the fourth sprint, the result of the sprint was the system design, reader login, and registration components. The fifth sprint's outputs were Reader's Loans and Loan History. In the sixth sprint, the Product Increment was Book Reservation and Availability of the work in the library. The team created a Search for books or documents in the seventh sprint. In the eighth sprint, there was an output-pom FAQ, i.e., an overview of frequently asked questions. In the penultimate ninth sprint, it was decided to complete the Filtering of records. In the last tenth sprint, the team's output was the Evaluation of books, i.e., Reviews and Recommendations of books based on previously borrowed and searched ones. The overall Product Increment was a mobile application.

In terms of processes, four essential project management ceremonies were implemented to help the team identify better goals, structure, and activities that best suited each of them. These four types include kick-off meetings (Sprint Planning), status updates (Daily Scrum), stakeholder reviews (Sprint Review), and project reviews (Sprint Retrospective). As for Planning, the goal was to create a goal for a given sprint (Sprint Goal) and create a Sprint Backlog, which is the second artifact of Scrum. The first is the Product Backlog. It is a list of tasks that must be completed for a given period that has been set or a time iteration, i.e., a sprint.

Sprint from the Scrum Master's point of view

The development team included developers who had never heard of the Scrum framework. It's why the Scrum Master explained to them what Scrum is, why individual ceremonies are essential, he explained why he was introducing an agile management method that was not only effective for the team but also for the customer, how tasks are recorded in Jira, what different benefits this agile management contains. He showed them that the position of Scrum Master had its meaning in terms of, for example, solving problems or relieving them of communication with the customer.

Sometimes he did a few things for someone else to teach them the process or to guide them. Sometimes he took on the role of Product Owner to replace her during her absence and thus tried another position as Scrum Master. So he also communicated with the customer and determined some priorities that the team worked on. Then he chose the approach, "*don't catch fish for a person, but teach him to fish*". When you give him a fish, you feed him for a day, but when you teach him to fish, you feed him for a lifetime.

If someone had a problem during the sprints, they contacted the Scrum Master using the Slack tool, and then they arranged an online meeting where they discussed the issue and agreed on a solution. But there were also problems where only the Scrum Master was informed, and he resolved the issues without needing a meeting. He always tried to solve and eliminate the problem in the shortest possible time so that the sprint was not prolonged and the team did not lose time. But it was not always easy and possible. The developers' problems turned out to be the most serious. A significant risk for this project was the new library-information system, which the Ján Bocatia Public Library wants to apply. Some institutions have already started using Tritius. In the case of the library, not everything went entirely smoothly. Communication with Tritius employees was the most critical. Tritius is not a modern system that

allows you to connect to an application. It does not offer the possibility of connecting to the system via API protocols. It uses old protocols, in which it isn't easy to search as well as to update the given protocols. Likewise, after a discussion with the owner of the library system, a necessary update of the system is not planned soon so that it is possible to connect third parties, e.g., mobile applications.

The Scrum Master tried to conduct the meetings so that as many team members as possible participated, if possible, all of them. Before each session, he also wrote to the team in Slack so they would not forget the meeting. He taught the team about responsibility and the importance of announcing their absence. If someone couldn't attend the meeting, he assigned another person to answer for him, so the team knew what the given team member did yesterday, what he would be working on today, and if he had any problems. It made these meetings so much easier. The whole team met only once a week, as it suited everyone and was enough. If there was any minor issue, they reported it to the Scrum Master using Slack.

Sprint Planning from the point of view of the Scrum Master

First, an initial project meeting (i.e., Sprint Planning) was held. It was a meeting where the whole team, including designers, developers, and marketers such as Scrum Master and Product Owner, came together. It is true that for a 1-month sprint, the duration of this meeting should not exceed 8 hours. Since the beginning of the project was not risky, it was not necessary to choose a sprint shorter than two weeks. The Scrum Master set the sprint to 2 weeks so that Sprint Planning could take a maximum of 4 hours. Our Sprint Plannings always lasted a maximum of 2 hours. Never has Sprint Planning taken longer. If Sprint Planning takes less time, but everything that is needed is covered, it can be done even faster. This condition was met every time, and tasks were always selected for the next sprint. However, it should never take longer than 4 hours in a 2-week sprint, so that the project is not delayed and the team does not waste time. It is such a time limit that the Scrum Master followed so as not to exceed it. Sprint Planning was planned as soon as the team was ready to go into planning mode. This meeting served as the formal start of project planning. The project kick-off meeting was the first meeting where the Scrum team came together to gain a common understanding of the goals and scope of the project and to understand each person's roles within the group.

The meeting started with a short introduction. The Scrum Master gave each team member approximately 10 minutes to introduce themselves and their role. After that, the project's background was explained for about 5 minutes. It included details such as how the project came about and why it is essential. This time was also used to set a common vision. Another 5 minutes were spent sharing the goals and scope, which referred to the boundaries around the project. It included clarifying what work was considered in-scope work and what work was considered out-of-scope work.

This meeting was also a good place to share the target launch date and highlight any important milestones the team needed to be aware of. Once the goals and scope were discussed, it was time to discuss the roles of all team members. It took about five minutes to ensure that everything was clear to everyone responsible for what worked throughout the project. Afterward, cooperation and how the team will work together on the project were discussed. Tools that served as a common source of information for the group were also explored. For example, a project plan created in a spreadsheet or work management software, e.g., Jira Software. In addition, it was determined how the team would communicate with each other, such as through daily e-mail updates, a team chat room, or weekly online team review meetings via Google Meet. Approximately 10 minutes were devoted to this topic. Once everything was ready and the project details discussed, it was time to discuss what would happen next. Over the next 10 minutes or so, expectations were set with the team members for what was to come. This was also used to clarify the next procedure for each team member, i.e., so that members know how to proceed. Finally, about 15 minutes were allocated for questions from the team. It was a chance for the Scrum team to gain insight into some of the topics we've discussed. So this kick-off meeting covered introductions, project background, goals and scope, tasks, collaboration, and what's next, but mostly time was reserved for questions from the group at the end.

All members of the Scrum team are invited to this initial meeting. During the kick-off meeting, team members learned more about how to contribute to the project and gained a deeper understanding of how the team would work together to achieve the project's goals. Stakeholders were also invited to have an opportunity to understand the project plan at a high level, share their perspectives and ensure that everyone is on the same page. Some meetings were more time-consuming, but some were less demanding when it was enough to send a quick e-mail or use chat with team members. After the program meetings were completed, this information was documented in a meeting agenda template and sent to participants a day or two before the meeting. The Scrum Master led most of this meeting.

Daily Scrum from the point of view of the Scrum Master

The Sprint started, the Sprint Planning was completed, the Sprint Backlog and Sprint Goal were available, and finally, the Definition of Completion was determined (i.e., it is generally understood that it is known when the task will end, what it should contain, and everyone understands it). The whole team usually met every day at this meeting, but it was a little different for us. The Scrum Master set Daily Scrum meetings once a week, where it was said who is working on what - so that it is not time-consuming for the team members. Then a meeting was held with the team, including the Scrum Master, Product Owner, designers, and developers. This meeting took place online via Google Meet and lasted no more than 15 minutes.

Sprint Review from the Scrum Master's point of view

Like all the previous ceremonies, this meeting was also limited in time, and it should last a maximum of 2 hours for a 2-week sprint. In our case, this meeting still lasted approximately one hour. All Sprint Reviews were conducted online using Google Meet, and the tools used in these meetings were Figma, where the designers showed their progress, and Xamarin, where the developers worked.

It was convenient for the client to meet with the team twice a month. The project was presented to the project sponsor, although only a small part was made. The client did not like the colors that were used. So it was recorded, and the Product Owner tried to work it into the next sprint. Therefore, it was essential to teach the customer to talk only at the Sprint Review so that the scope does not have to change during the sprint, i.e., so that during the sprint it was not needed to solve issues how to change the requirement, where and how it will be placed. Everything had to have its sequence. The Scrum Master and Product Owner explained to the customer that once the 2-week sprints were agreed upon, every two weeks, he would come to the Status and Sprint Review, and the customer

would see what had been done so far and provide feedback on what he wished to change. Subsequently, comments will be worked on.

Effective team communication is necessary to ensure the successful outcome of the project. The Scrum Master is the one who connects the team to the information they need. Throughout the project, he served as the main resource for the team when it came to communicating and clarifying goals, progress, and updates. There are many tools for sharing with the Scrum team, but Google Meet was used for online meetings. If a team member had a minor problem with a task or with any team members, they contacted the Scrum Master using Slack. The lengths of those meetings were not always as long as planned. For example, if the meeting was supposed to last 2 hours, it did not mean it lasted 2 hours. It was just the planned maximum possible length of the meeting. If the discussion at the meeting was finished in 20 minutes, there was no need to extend the meeting. The length of the work meeting depended only on the work team. The purpose of the Sprint Review needed to be fulfilled. This meeting should not last more than 2 hours to avoid disrupting the sprint later.
Sprint Retrospective from the perspective of the Scrum Master
Retrospectives should take place throughout the project's life cycle. Therefore, they were implemented after the main milestones, i.e., sprints, and after the project completion. As for the time limit, the Sprint Retrospective should not last longer than one and a half hours for 2-week sprints. This time limit was always respected, and the Retrospectives did not last even half an hour. These meetings were held online using Google Meet. In advance, the Scrum Master prepared a PowerPoint presentation, which he then shared during the meeting.

Agile is still relatively new in Košice, and some companies are now switching to it. Many people are fed up with it, especially developers, who until now have taken a task, made it, handed it over, and didn't have to deal with any of the processes around it. But because the Scrum Master showed the benefits and admitted that it is simply impossible to introduce Scrum hard because this alone is not agile. Still, it is necessary to introduce some agile management methods that will be effective for the team and the customer. It meant being flexible towards the client, i.e., the customer. He told the group that he chose Scrum because he saw the benefits. For example, in the Daily Scrums, he saw the benefits in planning, saw the benefits in the fact that it relieves the team of communication with the customer, saw the benefits of the Product Owner that he writes down the User Story and the team he just adds that they will already have these things, he saw the benefits in that our team will grow, the amount of work will be more significant, etc.

The retrospective was also resolved because not everything in the team always went as it should, and the Scrum Master wanted to help the team. He needed to write it down somewhere and the whole team for it. When he had outputs, he knew how to do something with them. He said two sentences, and the team was already listening. Because it was not ordered, but it was something that he wanted to do for the Scrum team. That was the role of the Scrum Master, to be human, explain everything, and lead.

At the end of each sprint, the Scrum Master's task was to prepare an exciting presentation and lead meetings such as Sprint Review and Sprint Retrospective. It was essential to know who felt how during the sprint so that he could change it in the next sprint. Not a single Retrospective, it turned out that anyone was disappointed or frustrated by the given sprint. Mostly it was either neutral or the team was satisfied. The main goal of the Retrospective is to answer three essential questions.

The first question is: "*What is done well?*". He wrote all the answers from the team directly into the presentation so that everyone could see it and compare it with the previous sprint. The designers managed to make a draft of the project, and the teamwork was successful. Team members constantly communicated with each other, helped each other, and were not afraid to say or ask anything. The team was greatly helped by the training, where the Scrum team learned how to proceed in case of any problems, and above all, there were no quarrels in the group. There was a perfect atmosphere, which was contributed not only by the Scrum Master but also by the whole team. He tried to motivate them all the time, either in their work or talking.

The second main question in the Retrospective was, "*What didn't work for us?*". The main and probably the most significant long-term problem was communication with Tritius. Even so, the Scrum Master always tried to solve this problem, especially to solve it as soon as possible. But it was not always easy. Sometimes Tritius did not answer the question directly, and our requests could not always be fulfilled. Moreover, this new technology did not suit us due to the lengthy creation of API protocols and the like.

The last point of the Retrospective is the third question, i.e., "*What needs to be improved?*". Improving communication in the team and in contact with Tritium is necessary. One of the possible risks that could have affected the project was that access would not be obtained.

4. RESULTS AND DISCUSSION

In the calendar year 2021, a team was created for the Book Avatar project, in which the mobile application development took place using the agile Scrum framework. In this part, the impact of the implementation of the agile framework on the development of a mobile application for the library is evaluated from three points of view:

- Progress of sprints during the project. Application features. Problems during the project.
- What were the tasks of the Scrum Master in the Book Avatar project?
- Our recommendations result from the implementation of the project.

4.1 Progress of sprints during the project

One sprint should not be longer than one month and shorter than one week. The length of the sprint was set to 2 weeks, i.e., the most common sprint length. A total of ten sprints had to be performed. The Sprint was divided into four ceremonies: Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective. The sprint alignment occurred in the 3rd sprint, which means that the first two sprints did not end as expected, unlike all other sprints. Only the design was developed for the first three sprints, but from the 4th sprint, the developers started working on the mobile application development. The project team started working on the highest priority tasks and then continued with the lower priority tasks. So, in the 4th sprint, the developers developed the Reader Login to the application and the New Reader Registration; in the 5th sprint they developed the Reader's Loans and Borrowing History; in the 6th sprint, they created the option of Reservations; in the 7th sprint the Availability of the work in the library, in the 8th sprint Searching for books, in 9. Filtering records, and in the last 10th sprint, they created Reviews, i.e., evaluation of books.

Features that the team managed to make and are the content of the application:

- login,
- book borrowing,
- history of book borrowing,
- favorite books,
- recommended similar books,
- book search filter,
- overview of events,
- frequently asked Questions.

During the entire course of the project, there was only one problem in the team. The problem was the team members' problem; he lost motivation and felt that not every member was fulfilling his role in the group. He solved this problem by adding another meeting in the form of a Daily Standup, where he asked each team member to show their work from the day before our meeting and then say or show the work they were going to do on the day of the meeting. It turned out that each of the team members performed their tasks honestly. He tried to solve every single problem as soon as possible so that the tasks in the sprint were not simply moved to the next sprint, or he tried to eliminate the given problem if possible.

4.2 Tasks of the Scrum Master

The main activities of the Scrum Master included: teaching the team the theoretical part of Scrum in practice, preparing the schedule for the whole month ahead, planning and facilitating individual meetings, observing ceremonies, supporting and motivating the team to fulfill tasks, helping the Product Owner with maintaining unresolved tasks in the Product Backlog, preventing or removing obstacles to the team's ability to meet project goals, relieving the team of problems of a variety of nature, such as pointless meetings, communication and administration, protecting the group from disruptive external influences, and encouraging the team to grow personally and improve.

4.3 Recommendations

Based on our experience, the following recommendations are written regarding Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective.

During Sprint Planning, specific User stories were selected that would be done in that one sprint. This meeting served as the official start of the project and as a way to align the team's understanding of the project's goals with the actual plans and procedures. The goal of Planning was to come up with everything that needed to be done for the sprint and explain everything in 1 day or 2 hours. At this meeting, the Product Owner answered questions and said the priorities on which User Story he would most like to be worked on. Explaining the Product Owner's vision for the given sprint was essential, e.g., it is a priority from the customer side, so the Home Page is available at the end of this sprint. Then he explained what the Home page should contain. The Product Owner always explained everything to us well so that everyone on the team had the same understanding of the tasks. It did not happen that someone imagined two different things in the team under one term. It was the purpose of that planning. The group chose the tasks they could complete in the first sprint, and the team members explained them to each other in such detail that everyone understood it and that nothing overlapped. Therefore, there was a general understanding. All team members were talking about the same thing. At the same time, during the Sprint Planning, the team members divided the tasks among themselves in the group. It means that by the end of the Planning, everyone already knew their roles and responsibilities.

Another important thing in sprint planning was the task description, which was written as a fairy tale/story (i.e., who wanted the task, what the task was supposed to do and why). The job description further included a description of the acceptance criteria. If the criteria were in line with the user's requirements, the given user story could be considered finished. They had to be written in detail so that everyone could understand them so that it was clearly defined where the end was. When this was done, the User Story was finished. The Sprint Backlog was visible and transparent to all team members. For this reason, the project team decided to use Jira Software for sprint planning.

Based on experience, the following recommendations were proposed for the next project:

- during sprint planning, team members should have tried to explain each task in as much detail as possible, including acceptance criteria;
- it should be clear and distinct when a task is completed and how a new team member can judge it, for example, how long it takes to complete this task, etc.;
- keep mandatory participation for all team members, even if, for example, there are more designers or developers on the team;
- divide individual tasks into the smallest possible parts, so that team members are more motivated to work on individual tasks, including sprints. If those team members only see, for example, five larger tasks on the board in the "*To Do*" column, they may not be as motivated to do the individual tasks, either in terms of the difficulty of those tasks or in terms of free time, such as those tasks, but smaller ones that only take a few minutes to complete.

The content of the Daily Scrum meeting was for the whole team to answer the following three questions: "*What did I do yesterday? What will I do today?*" and "*Do I have a problem*?". The team members needed to meet consistently at the same time in the same place, so it became a habit for them. Nothing else was discussed at our Daily Scrum. If there were any problems in the team, I solved them from the position of Scrum Master. Team members talked about problems right from the start as they arose. The group communicated nonstop.

The primary goal was to align the team with updates, progress, challenges, and next steps. The entire project team had to know who was doing what. Not because of control but because the development is fast, five months went by like water. There were many tasks and questions, and everyone just needed to know what everyone was doing so that the group members could function as one team; since there are no individuals in Scrum, everyone pulls together. The Scrum Master was interested in the problems, and the team as developers was interested in who did yesterday and what they will do today. First, the design was done, and then the programming started. First, the backend was programmed, and then the frontend (i.e., the frontend was connected to the backend). It often happens that there is some discrepancy, that the frontendist waits behind the design or the backendist wait behind the design, then the frontendist stays behind the backendist; simply these dependencies and so on, then they are more visible if the team participates in those Daily Scrums. In our case, there was no need to wait for the design. The result of those Daily Scrums was a list of all the problems that could be solved from the Scrum Master position. One of the critical responsibilities of the Scrum Master was to be aware of the status of the project at any given time and to ensure that others were informed or knew where to find the latest information.

During this meeting, a task update status was discussed in the team, where members discussed the status of the most urgent tasks, how many tasks have been completed, how many tasks are left unfinished, and the schedule's status. There was also discussed whether delays occurred, what was not happening according to the plan, or whether the pace of solving individual tasks was being followed. Further, current and expected changes, risks, resource issues, supplier issues, etc., were discussed. The status meeting was an essential project tool that kept the project on track.

For the next project, it is recommended that the other Scrum Masters use a relatively fixed schedule for this meeting and stick to the scheduled time. A fixed schedule will keep the team engaged. These status meetings would be maintained regularly. If Daily Scrums are routinely organized, these meetings will be beneficial because they will allow recognizing milestones, sharing information, and expressing concerns to the Scrum Team. From the project solution knowledge, it is recommended to organize these meetings every day, as it is written in Scrum, so that it does not happen that some members feel that not everyone is doing their work. There was a problem in this project because some team members felt that not everyone was doing their job in their position. This problem was solved by requiring all team members to post daily in Slack, where a group was created. In Slack, the activities of individual team members from the previous day were recorded, as well as what was done on the current day, and problems, if any, were also recorded there. If they knew how to document it, they uploaded the file or photo to our Slack group. That was enough to solve the problem.

The Sprint Review was a meeting at the end of each sprint in which the entire team (i.e., Product Owner, Scrum Master, and Development team) participated. The customer, that is, the client, was also invited. It was the only meeting where the customer had to come, and he needed to be there. We, as the Scrum Master, welcomed the customer, then the Product Owner presented to him what was done during the sprint. The essence was to show what worked for us. At the end of the meeting, the Scrum Master asked what the customer liked or didn't like and collected feedback (i.e., the most important thing the project team could get).

For further projects, a recommendation was made that the Scrum Master, together with the Product Owner, prepare, for example, a presentation in PowerPoint, where there would be graphs and examples of increments, i.e., parts of the products they created together with the Scrum team. Till now, they preferred to make a live demonstration of the product and its functions. But, again, this will reflect their control and expert knowledge of the product.

There is no better tool for reflection and ensuring continuous improvement than conducting a Sprint Retrospective. It was an internal meeting where the project team tried to find improvements to their processes (e.g., a discussion about using another programming language to improve the implementation of tasks, etc.). It was investigated which procedures should be continued and which could be improved. The meeting helped the team to get feedback about the whole research group, both from a professional and personal point of view (i.e., whether the team members worked well together, the meetings that were held were sufficient, the team members shared enough information, the technologies that the team had at their disposal were not outdated, the information provided did not cause any problems, or how things could be done better).

The Scrum Master listened to it all, found out the possibilities, and most of all, acted. The Scrum Master built a culture in his team, strived for continuous improvement, enabled team members to understand different perspectives within the Scrum team and thus better understand each other, facilitated collaboration, and increased team productivity. He asked for feedback on any aspect of the project, from planning, implementation, communication, and team dynamics. When the team saw the response that the Scrum Master or Product Owner wanted to help the team, it helped morale, speeded up development, and there were much fewer conflicts in the group. It wasn't a problem if something couldn't be solved, but it was also good from a Scrum Master's or Product Owner's point of view not to promise something they couldn't deliver. It was still necessary to explain everything to the team and say why not. They didn't say no and didn't turn on their heels and leave, but they always tried to give the team reasonable reasons. If they gave reasons to the team, they were more willing to listen. No one listened to them if they had just ordered something from the group. So everything still had to be justified. It helped improve communication and teamwork significantly. Also, this was true for any ceremony. They always asked the team why they were going to do the given ceremony, the benefit of the given ceremony, and so on.

Under no circumstances was the customer present at this meeting, only the team (i.e., including the Scrum Master, Product Owner, designers, and developers). The Scrum Master always led this meeting. The content of this meeting was to ask two questions: "*what was good?*" and "what could be done differently" (i.e., he never asked what was wrong). It was important that everyone could express themselves.

Nevertheless, when one or more issues are found in the Scrum Team during the Sprint Retrospective, it is recommended that the discussed change be made anyway. Because then people will not want to participate in retrospectives if they feel that their feedback is not fully considered and implemented. Based on the knowledge gained, it is also recommended that Scrum Masters explain all their actions. When you tell the team that every day at 8:00 a.m., you will have a meeting for 15 minutes and nothing more, half of the people will fall asleep, and the other half will not. But if you explain why it is like that, you give them the possibility that you will somehow change it if it doesn't work. As a result, the team will be more willing to try and do it.

5. CONCLUSIONS

Based on innovation in the field of libraries and the introduction of digital products, the Scrum framework was implemented in practice, i.e., in the Book Avatar project to develop a mobile application for the Jan Bocatio Public Library. This application simplifies and streamlines the process of borrowing books. With the help of the mobile application, the library can get closer to the readers of the books and news about events, new books, book reservations, and their subsequent borrowing.

The paper described the process of implementing the agile Scrum framework into the Book Avatar project for developing a mobile application and the team's satisfaction with the implementation. It was examined whether the team was motivated enough, how problems were solved in the group or how the team was satisfied with the set number of meetings during the sprints (i.e., the project duration). In addition, an evaluation was made of the optimal sprint setup, whether these setups were sufficient or not, and how satisfied the library was with the application.

Based on our experience from the point of view of the Scrum Master, Scrum ceremonies are described when the project can fail due to various errors and ways to prevent these errors.

Based on the evaluation of individual ceremonies (from individual sprints) with the implementation of the agile framework in the team, it is desirable to follow the rules for all ceremonies in other projects. For example, for one of the following projects, it is necessary to prepare a schedule with individual ceremonies, an exact time frame, and an agenda and ensure that the team is motivated, tasks will be done and completed on time, and the project will not be delayed. No other problems will arise, as in this Book Avatar project.

In the following research, it would be appropriate to supplement the mobile application with the following functions: Reading challenges (challenges) for greater reader motivation, Online pass in the application (using a barcode or QR code), Ranking of the most read books (in general), News, possibility to set the application's day or night mode (light or dark appearance of the application), Book reservation or document, Setting notifications and the opportunity to download the application also via the iOS operating system.

It is about the first library application in Slovakia, owned by the Jan Bocatia Public Library. The application was named Bookatius by combining the words "Book" and the "Bocatius" part of the library name. It's a natural but apt name for that. The project turned out successfully and was handed over on time. The client, the Public Library of Jan Bocatia, was satisfied with this application. Unfortunately, the application cannot be downloaded through the Store (Google Play) yet, as it has not yet been officially launched. The official launch date of the app is tentatively scheduled for March 2024.

References

1. KOLLNIG, K.; SHUBA, A.; BINNS, R.; KLEEK, M.V.; SHADBOLT, N. *Are iPhones Really Better for Privacy? A Comparative Study of iOS and Android Apps;* arXiv:2109.13722v4, 2021. Dostupné z: https://doi.org/10.2478/popets-2022-0033
2. LEITH, D.J. *Measuring the Data iOS and Android Send to Apple and Google; Mobile Handset Privacy;* International Conference on Security and Privacy in Communication Systems, 2021. Dostupné z: https://doi.org/10.1007/978-3-030-90022-9_12
3. ANGGREAINY, M.S.; SULAIMAN, A.P.; MATHEW, C.; TIRTA, K.E. *Approach to Problem Solving and Its Relation to the Scrum and Agile Software Development Method; Task Management;* Computer Science Department, School of Computer Science, Bina Nusantara University, Indonesia; Jurnal Ilmiah KOMPUTASI, 2021, Volume 20, No: 4. Dostupné z: https://doi.org/10.32409/jikstik.20.4.2871
4. JÚNIOR, P.S.S.; BARCELLOS, M.P.; DE ALMEIDA FALBO, R.; ALMEIDA, J.P.A. *From a Scrum Reference Ontology to the Integration of Applications for Data-Driven Software Development;* Information and Software Technology, 2021, Volume 136. Dostupné z: https://doi.org/10.1016/j.infsof.2021.106570
5. JONDYA, A.G.; ANWAR, H.I.; HERMAWAN, M.R.; DERU, T.A. *A Fantasy Puzzle Adventure Game with An Approach to Agile by Applying Game-Scrum Methodology;* Developing “EMBER”; JURNAL EMACS (Engineering, Mathematics, and Computer Science), 2022, Vol.4, No.2. Dostupné z: https://doi.org/10.21512/emacsjournal.v4i2.8428
6. ALAZZAM, M.B.; ALASSERY, F.; ALMULIHI, A. *Development of a Mobile Application for Interaction between Patients and Doctors in Rural Populations;* Hindawi, Mobile Information Systems, 2021. Dostupné z: https://doi.org/10.1155/2021/5006151
7. GOMERO-FANNY, V.; BENGY, A.R.; ANDRADE-ARENAS, L. *Prototype of Web System for Organizations Dedicated to e-Commerce under the SCRUM Methodology;* (IJACSA) International Journal of Advanced Computer Science and Applications, 2021, Vol. 12, No. 11. Dostupné z: https://doi.org/10.14569/IJACSA.2021.0121189
8. CANEDO, E.D.; CERQUEIRA, A.J.; PAPADOPOLIS, A.V.; DE ARAUJO, A.P.F. *Application of Design Thinking for Requirements Elicitation in Mobile Applications;* Proceedings of the 53rd Hawaii International Conference on System Sciences, 2020.
9. SETIAWAN, R.; SUPRIATNA, A.D.; HUDAWIGUNA, S.; ROJI, F.F. *Electronic culinary reservations based on Android with the Scrum methodology and Firebase database;* IOP Conference Series: Materials Science and Engineering, 2021. Dostupné z: https://doi.org/10.1088/1757-899X/1098/5/052091
10. FITRIANA, G.F.; PRASETYO, N.A. *Rice Planting Calendar Application Development using Scrum;* Indonesia; IJCCS (Indonesian Journal of Computing and Cybernetics Systems), 2022, Vol.16, No.2. Dostupné z: https://doi.org/10.22146/ijccs.70155
11. CHANDRA, A.Y.; PRASETYANINGRUM, P.T.; SURIA, O. *Virtual reality mobile application development with scrum framework as a new media in learning English;* Universitas Mercu Buana Yogyakarta, Bantul, Indonesia, 2021, Vol. 15, No. 8. Dostupné z: https://doi.org/10.3991/ijim.v15i08.19923
12. RAMOS-ROMERO, A.; GARCIA-YATACO, B.; ANDRADE-ARENAS, L. *Mobile Application Design with IoT for Environmental Pollution Awareness;* Facultad de Ciencias e Ingenier´ıa; (IJACSA) International Journal of Advanced Computer Science and Applications, 2021, Vol. 12, No. 1. Dostupné z: https://doi.org/10.14569/IJACSA.2021.0120165
13. SAMA, H.; DARVIN, D. *Developing Fixher Using Scrum Model.* Information System Study Program; Universitas Internasional Batam; Journal of Information System and Technology, 2021, Vol.02, No. 02.
14. AKLANI, S.A.; LIM, D. *Mobile–based donation application design and development using the SCRUM method.* Faculty of Information System, University Internasional Batam, 2022.
15. SIAHAAN, M.; BIMANTORO, A. *Design, and development of useful item donation applications with SCRUM method.* Faculty of Information System, University Internasional Batam, 2022.
16. AGUIRRE, J.F.L.; ANDRADE-ARENAS, L. *Geolocation Mobile Application to Create New Routes for Cyclists.* University of Sciences and Humanities, Faculty of Sciences and Engineering, Lima Peru; (IJACSA) International Journal of Advanced Computer Science and Applications, 2022, Vol. 13, No. 2. Dostupné z: https://doi.org/10.14569/IJACSA.2022.0130280
17. IMRON, A.S.; RAHARJO, T.; HARDIAN, B.; SIMANUNGKALIT, T. *A case study at poultry startup in Indonesia; Gamification to improve scrum adoption;* Universitas Indonesia, Department of Information Technology, Jakarta, Indonesia, 2022, Vol. 100, No. 20.
18. MAYOR, J.; LÓPEZ-FERNÁNDEZ, D. *Virtual Reality Serious Video Game to Learn Scrum; Scrum VR;* Departamento de Sistemas Informáticos. Spain, Applied Sciences, 2021. Dostupné z: https://doi.org/10.3390/app11199015
19. BOZA-CHUA, A.; GABRIEL-GONZALES, K.; ANDRADE-ARENAS, L. *Implementation of a Mobile Application for Blind Students; Inclusive Education;* (IJACSA) International Journal of Advanced Computer Science and Applications, 2021, Vol.12, No. 11. Dostupné z: https://doi.org/10.14569/IJACSA.2021.0121189
20. IPARRAGUIRRE-VILLANUEVA, O.; SIERRA-LIÑAN, F.; CABANILLAS-CARBONELL, M. *Location-based Mobile Application for Blood Donor Search;* (IJACSA) International Journal of Advanced Computer Science and Applications, 2022,

Vol. 13, No. 4. Dostupné z: https://doi.org/10.14569/IJACSA.2022.0130418
21. ASHARI, I.F.; ZUHDI, M.F.; GAGAMAN, M.T.; DENIRA S.T. *Kolepa Mobile Application Development Based on Android Using SCRUM Method;* Journal of Applied Informatics and Computing (JAIC), 2022, Vol.6, No.1.
22. LEON-AYALA, R.; GOMEZ-CORTEZ, G.; ANDRADE-ARENAS, L. *Mobile Application Aimed at Older Adults to Increase Cognitive Capacity;* (IJACSA) International Journal of Advanced Computer Science and Applications, 2021, Vol. 12, No. 12. Dostupné z: https://doi.org/10.14569/IJACSA.2021.0121297
23. SENTOSA, J.; MAHARINA, M.; ZONYFAR, C. *E-Ticketing System and Integration with Third Parties Scrum-Based; Technical Information*, STIMIK Likmi, Universitas Buana Perjuangan Karawang, 2021, Vol.2 No.2.
24. MORANDINI M.; COLETI, T.A.; OLIVEIRA, E. JR; CORRÊA, P.L.P. *A survey for analyzing results for development teams; Considerations about the efficiency and sufficiency of the utilization of the Scrum methodology;* Computer Science Review, Brazil, 2021, Volume 39. Dostupné z: https://doi.org/10.1016/j.cosrev.2020.100314
25. YOUNUS, A.M.; ABUMANDIL, M. *Evaluating the role of scrum methodology for risk management in information technology enterprises;* Journal of Information Technology and Computing, 2021, Volume 2. Dostupné z: https://doi.org/10.48185/jitc.v2i1.159
26. CARBONE, R.; BARONE, S.; BARBARESCHI, M.; CASOLA V. *Agile Development in Safety-Critical Software Systems; Scrum for Safety;* International Conference on the Quality of Information and Communications Technology, 2021. Dostupné z: https://doi.org/10.1007/978-3-030-85347-1_10
27. KOMAR, O.A.; CHUCHALINA, Y.M.; KRAMARENKO, A.N.; TORCHYNSKA, T.A.; SHEVCHUK, I.V. *Agile Approach in Training Future Primary School Teachers for Resolving Complex Pedagogical Situation;* International Electronic Journal of Elementary Education, 2021, Volume 13. Dostupné z: https://doi.org/10.1007/978-3-030-85347-1_10
28. ROCHA, M.; BARBOSA, E.; GONZALEZ, C. *Agile Management for Industrial Engineering Capstone Projects – Overcoming the COVID-19 challenge;* Proceedings of the International Conference on Industrial Engineering and Operations Management Monterrey, Mexico, 2021.
29. ORLOV, E.V.; ROGULENKO, T.M.; SMOLYAKOV, O.A.; OSHOVSKAYA, N.V.; ZVORYKINA, T.I.; ROSTANETS, V.G.; DYUNDI, E.P. *Comparative Analysis of the Use of Kanban and Scrum Methodologies in IT Projects;* Universal Journal of Accounting and Finance, 2021. Dostupné z: https://doi.org/10.13189/ujaf.2021.090415
30. HAVAZÍK, O.; PAVLÍČKOVÁ, P.; RYDVAL, J. *Agile Game in Practice;* International Workshop on Model-Driven Organizational and Business Agility, 2022. Dostupné z: https://doi.org/10.1007/978-3-031-17728-6_3