Low-code software development

The platform is Microsoft PowerApps

Tarmo Urrio

Thesis, MA

July 2022

Engineering

Information and communication technologies

Jyväskylän ammattikorkeakoulu
University of Applied Sciences

Urrio, Tarmo

Low-code software development:
The platform is Microsoft PowerApps

Jyväskylä: Jyväskylä University of Applied Sciences. July 2022, 40 pages.

Engineering. Information and communication technology degree program. Thesis AMK.

Language of publication: Finnish

Permission to publish on the open web: yes

Summary

As working methods and environments change and remote working increases, the need for software produced for the needs of companies is constantly growing. The need for new applications is great, and as the skills shortage in the software industry slows down the necessary growth and development of companies, the need for new skills is also increasing.
Low-code application development, which is growing in popularity every year, partly answers these challenges.

The aim of the thesis was to investigate Low-code application development, the reasons for its popularity and generalization, and compare the assumptions about development on these platforms and the advantages of the development method compared to standard application development. Microsoft's PowerApps was the application development platform to be studied more closely.

In the research, an example application was implemented, during which the implementation of the selected platform, its features, and the various aspects of the development taking place were monitored. The comparison looked at the areas involved in the development, and the process used to produce the application. The information and experiences gained from the research were compared to the experience of ordinary application development and development processes.

Based on the research, it was possible to state that the development on Low-code platforms largely corresponds to the assumptions set at the beginning of the research, and enables functional application solutions, especially for the internal needs of companies, in a very cost-effective manner. For a person with a background in software development, transitioning to Low-code applications and adopting the development method is very easy.

Application development on low-code platforms will play a significant role in the future as part of newly produced application solutions. With these solutions, the internal processes of companies can be made more efficient, as well as cost savings and speed up complex processes, by enabling the collection, display and management of data from several data sources, with the help of a single application.

Keywords

Low-code, Software development, Microsoft PowerApps

Other information (confidential attachments)

Description

Urrio, Tarmo

Low-code Software Development,
Microsoft PowerApps as a platform

Jyväskylä: JAMK University of Applied Sciences, July 2022, 40 pages.

Information and Communications. Degree program in Information and Communications technologies. Bachelor's thesis.

Permission for open access publication: Yes

Language of publication: Finnish

Abstract

As working methods and environments change and working remotely increases, the need for software produced for business needs is growing. The need for new applications is significant, and as the shortage of experts in the software industry slows down the necessary growth and development of companies, the need for new experts is also growing. Low-code application development, which is increasing in popularity year by year, contributes to these challenges.

The aim of the thesis was to study Low-code application development, the reasons for its popularity and prevalence, and to compare assumptions about the development on these platforms and the advantages of the development method compared to ordinary application development. The application development platform that was studied in more detail was Microsoft's PowerApps.

An example application was implemented in the study, which was used to monitor the implementation and characteristics of the selected platform, as well as the various aspects of its development. The comparison focused on the development aspects and the development process. The data and experiences obtained from the study were compared with the experience of ordinary application development and its processes.

On the basis of the study, it was found that the development on Low-code platforms largely corresponds to the assumptions set at the beginning of the study, and enables efficient application solutions, especially for the internal needs of companies, in a very cost- effective manner. For a person with a background in soft-ware development, the methods and transition to Low-code applications are very effortless to learn.

Application development on low-code platforms will play a significant role in the future as part of the new application solutions produced. These solutions can be used to streamline companies' internal processes, both to generate cost savings and to speed up complex processes, by enabling data collection, display, and management from multiple data sources with a single application.

Keywords/tags (subjects)

Low-code, software development, Microsoft PowerApps

Miscellaneous (Confidential information)

Contents

Patterning

1 Less code, more value

The usual software development process is often long, binds a large group of developers to a single project to a couple and can become expensive for the customer. Possible problems or corrections of the produced program needs are often essential. A typical software development model may include up to 16 different stages (The Low-Code Development Guide Nd) and contains a considerable amount of possible variables that cannot be estimated in advance. The software production costs can be influenced by these rise significantly higher than the original plan.

A good example of this is the Aster patient information system designed for four Finnish social security regions. a system, the implementation of which eventually failed partly also due to increased development costs. In the end, the final bill, which tripled the original contract amount, was too much for the subscribers and such a big reason for ending the project before the product is finished. (Lindroos, K & Björklund, S 2021.)

In normal software development, cooperation with the customer can be challenging, because the project there can be several stakeholders involved, and the project's current state of production becomes precisely clear to the customer only when the product is ready for testing or presentation. Before this, the customer's perception of the status of the project's development can be based, for example, only on the situation overview of the project's client managers si or small introductions about the state of the program, and is largely dependent on what the producer tells. Develop tys ties the company's resources to the project, and development cooperation with the client is often limited, because often the customer does not have technical knowledge of the necessary background systems or the development process processes.

What if software development were easier, faster, and the rising costs were just the result about adding new features to the finished product? What if the customer could see the products- of the current state of the program at any moment, and at best even participate implementation other than by participating in the definition of the program's requirements? To these problems corresponds to low-code software development, which is increasing in popularity every year and are included as part of modern software solutions.

The increased popularity is influenced by several different factors, of which even the most important role is played by the the small amount of work used for the development of the body. Low-code software development possible guarantees fast implementations, efficient integrations with other systems and data sources, and low-sheep's threshold way of creating software for different uses. The use of low-code platforms is based on trends, a large part of the future in software development (Vailshery, L. 2021). These vessels these provide the opportunity for programming, software design and implementation also in for people who don't necessarily have any previous background in software development.

Low-code platforms enable program planning and implementation with a simple user interface so that the developer does not have to create everything in the background for the program from the beginning damn program code. Utilization of low-code platforms reduces the amount of time required for program production meets development work, costs and enables direct cooperation with the application subscriber, with the product's development level constantly visible. The solutions produced in this way are lowered from the customer's threshold to choose the software solution that is specifically targeted to the company's needs product. Both the subscriber of the program and its implementer benefit from this, each in their own way ways. These solutions can also be easily connected to several different data sources, so you're ready to go it is possible to use the information in the application in a very versatile and ordinary way compared to software development much easier, by minimizing the connection of different interfaces the challenges brought by

The aim of this work was to investigate low-code software development, its advantages or disadvantages and always growing popularity on a general level. The work examines the factors affecting low-code development solutions, the general features of the platforms, the development that can be implemented with them, and examines them through an example application development on Microsoft's PowerApps platform. The work also compares the current general men-lipid from the subject area, and compares it with the experience experienced through the production of an example application there. The study excluded the comparison of different development platforms and the systems operating behind these telms operation.

The work sought answers through empirical research to the popularity of low-code, to how no-it is important to internalize the way and practices of development and how challenging it might be. Investigate in this paper, the implementation of an example application implemented on the PowerApps – development platform was examined and observe the process used for this, influencing factors and achieved results, ver- as a pre-set general opinion derived from the research data base.

2 Database

2.1 Standard software development

Normal software development is based on several different phases, where the software is built from from fiber through design, implementation and testing to the final product. Software to- one or more programming languages can be used for implementation, and development often takes place in several by a team of professional developers. Everything that happens in the program is visible or in the background, has been written into the program format either by hand or added to be used by the program in the pre- as a sign of a ready-made open source solution. Also, some sections of the program can may be, for example, from an external service provider, or used from publicly available ones program libraries.

The amount of resources required for the implementation of the program depends on the scope of the software to be implemented or depending on complexity, but at the same time options for countless different ones are also possible for solutions, fine-tuning of the program and precisely customized according to customer wishes for implementations when creating a program. Possibilities for different implementation methods and the freedom to implement the program code exactly in accordance with the developers' preferences also bring advantages and pos- minor disadvantages to the final product itself.

To ensure the quality of the software, a lot of testing is carried out for the produced program, and quality control, which also consumes resources and ultimately produces more costs thanks to the customer. Possible faults are tried to be screened from the program already in the initial phase and the customer lo- The product to be delivered to Bolta is officially ready for the final rounds of approval and for the customer after delivery. After this, the maintenance phase begins, during which the software product room is maintained and updated, possible malfunctions are responded to and the product is developed according to the needs of the battery until the product reaches the end of its life.

2.2 Low-code software development

As the name suggests, low-code is software development where code is needed as little as possible little or not at all. It is based on the often cloud-based so- to the vellus development platform, with which the program developer is offered the development of this type of software the environment needed for the hit, most often at the price of a regularly billed license fee. There are many platforms intended for low-code development, and their service providers and products change from well-known large companies to smaller solutions based on open source code.

Development takes place based on the user interface, either through an internet browser or on a computer. van software. The developer has a development environment at his disposal, in which the graphical user interface options offered by the development platform, such as different components or functions settings and other necessary tools to manage the visuals visible to the end user ten layouts and the data imported into them. The development platforms base their operations on various connected with which the program to be produced is combined to retrieve information from other systems in use from, for example, an Excel table or a separate database. Searchable information about these processed and displayed through freely selectable components in various created views, from which the requested information is more easily available or processed or presented as needed.

Low-code applications are often most commonly implemented for various internal needs of companies, such as cor- to use several different management tools or, for example, to collect data from several different data sources into one screen, which makes managing the necessary information much easier. Low-code enables continuous testing already while making the application, thanks to which possible misconfigurations rations can be noticed and solved faster even before they end up in program testing to the stage. Even with the small amount of program code required for the operation of the program itself, the way to create programs is also dolled up for people who have no prior experience in programming, by simplifying the process of creating a new program and the necessary know-how and tools ver- as a cog for standard software development.

2.3 No-code software development

The next step down from low-code in the amount of code needed is so-called no-code. With these platforms, the possibility to modify the background code controlling the program's functions has been removed, and these development platforms are significantly more limited compared to low-code development platforms. In the absence of modification of the background code, the user has no possibility to cause a situation, where the program would no longer work due to a change in the background code made by the user. No-code vessel in others, the lack of adding code becomes a limitation of development, the flexibility of these platforms the ability to implement solutions for more specific needs.

Development on these platforms is completely dependent on the limits provided by the options provided by the platform. Ra- the possibilities of the development platform implemented by the producers of that platform ultimately serve as det. If a feature is not designed for the platform, it is very likely that it is not able to implement at least enough different platform features by combining them. No-code the platforms are targeted more at users who have no prior experience in software development its background. These solutions are therefore suitable, for example, for business users who want to create no- made a simple application, for example, speed up the company's internal operations (Forsyth 2021).

2.4 History

The first low-code-like platform can be considered the Hypercard system developed by Apple conclusion from 1987. This made it possible to create programs similar to today's development platforms without writing concrete program code, but also enabling, if necessary, program man for editing the background code. (Lasar 2019.) Even at this stage, the target group was the new ke- developers who did not need to have previous programming experience. Development of this platform was discontinued after a little over ten years, after which the current popularity of the development method the beginning can be considered the appearance of the low-code concept in 2014. At that time, the well-known American the wealthy research company Forrester published an article discussing the new growing popularity the generalization of the development method and used the term low-code to name the respective development method and platforms to meow. (Tozzi 2021.)

The use of the term became more common as the platform's continued popularity grew and developed over the years, and it still is rose as a concept to its current level of familiarity. As the development method grows in popularity, various players, starting with the largest ones, have started to produce their own development platforms with which to respond to this to growing demand.

2.5 The future of software development

The solutions for rapid software development have increased in popularity year by year, and have worked increasingly well-known also among customers who order software. The software sector is a rapidly growing industry, and at the same time the need for those implementing these solutions is on the rise. However, according to Schoettle (2021), low-code development should not be misunderstood as standard programming as a substitute for hybrid development, but rather as an alternative that works alongside it. Both of them development methods have their own advantages, and it is up to the software subscribers and developers themselves to decide which the way is best suited for the software being produced.

As a low-code development method, it benefits from a developer familiar with the development work of the respective platform chicken condition in the implementation process. If the implementation of the program takes place, for example, internally within the company, and the people attached to the project do not have a programming background, nor the possibility to engage security involving a blue-chip software developer in the project, may be the best choice for the development platform suitable for no-code centralized platform. In low-code development, it is about programming skills, however- also considerably useful, as the programs created on the platform need additional program code to function in the desired ways or to limit the information to be searched. The development platform itself speeds up the moderately already the process of producing the program, but still does not complete everything for the user, but rather provides a framework on which to build the solution they need. Worry about this matte development platforms offer a significantly reduced starting point for producing a program, and due to the available guides and tutorials, you can develop your own low-code program help to catch up very quickly.

Companies often have people who recognize business needs and processes, but have to vat to review the information they need in their work either by hand or from large Excel tables. Low- by providing the know-how needed for software development, and enabling the production of the program itself to more and more people, growth is generated for industry players and companies that need software and development through digital solutions that support business. By easing the threshold for program- to mixed development, possibly also the role of these so-called citizen developers and participation as part of the creation of new solutions to speed up or improve the company's various business aspects of mints. (Lindström 2021.)

Due to the rapid growth, the benefits brought by the development method and the resulting popularity, it has been assumed, that by 2025, up to 70% of new applications to be developed could be implemented using low-code platforms, the corresponding figure for 2020 being 25% (Stamford 2021). Provided the current direction of software development and the adoption of platforms will continue at the same pace, is this a high-sounding reading quite possible.

2.6 Market situation

The demand for low-code solutions has rapidly increased over the past few years, and thus also the market there have been several different development platform solutions for them, from which the developer may have difficulty choosing the root a solution suitable for their own needs. Currently, the platforms' competitive advantage is the combination of possibilities for different data sources and systems, pricing, ease of use of the platform and scale volume according to possible needs. The competition between platforms is fierce, which adds to it also innovation, and thus brings new technical solutions to the market. As Figure 1 shows- yes, the market situation is going through a phase of rapid rise, low-code rapid, according to forecasts for popularity. The annual revenue of the development platform market is predicted to grow by more than 50% to one billion euros by 2027.

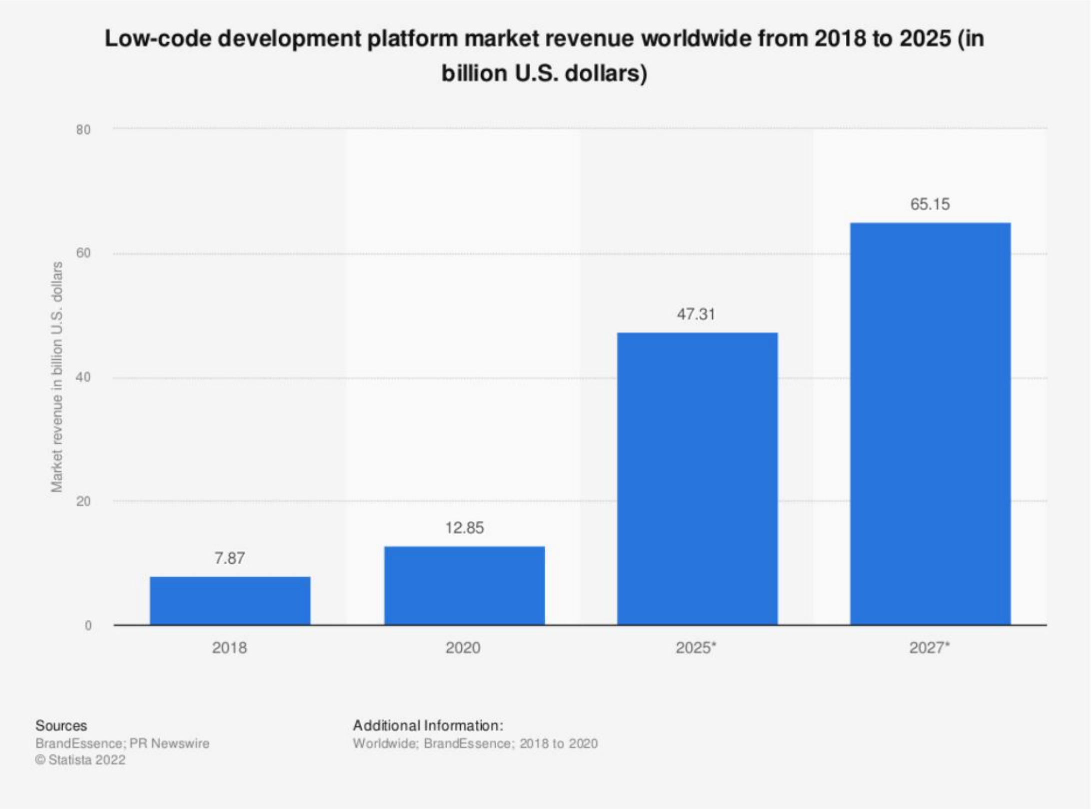


Figure 1. Predicted market revenue of low-code development platforms (Vailshery 2021)

There is a constant need for different software, especially for a specific purpose

on the increase. The response to this need is also partially slowed down by the shortage of skilled workers that plagues the industry. Regular-

in that software development, it is desirable that the employee developing the software already has experience

about the development processes themselves and the programming languages chosen for the implementation of the program.

Due to the skills shortage, it can be difficult to find employees needed for projects, which is why companies

the company may even fail to make an offer completely, and the project is often awarded to the company as it sees fit

Competent people with kemus can be found as needed.

Low-code is also partly an answer to the market need, which desperately needs skilled workers to

to implement new applications, especially for the internal needs of companies. As the digitalization of companies progresses

and the change in working methods brought about by the global pandemic that started in 2019 accelerated

essa, more and more companies have recognized the need for new internal tools and other solutions

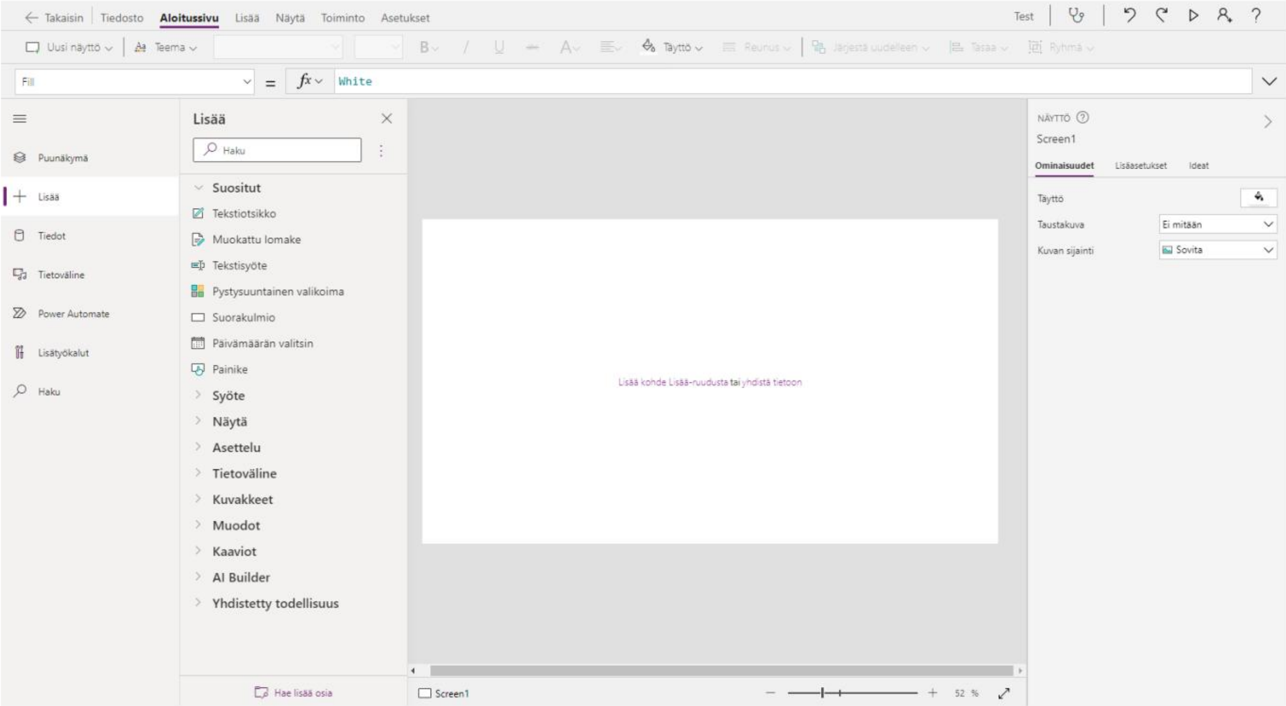
to make remote workers' everyday life easier.

2.7 How development takes place

Low-code development differs from regular development not only in the amount of code, but also in its visuality

due to the development method. The developer creates software from the user interface, where he has visible user

available components, connectable data sources and other tools necessary for development.



Fill = *fx* White

Täyttö

Taustakuva: Ei mitään

Kuvan sijainti: Sovita

Figure 2. Start window of the PowerApps development platform

The development progresses by adding the necessary views, to which various components are added, such as text fields table or tables. The necessary additional logic can be added to these components, so for example the text entered by the user can be set as the user's username or the table information can be obtained entered into the database. Several views can be created for the application, including different functional mouths and transition possibilities between them, depending on the user's activity. The necessary external edits of an outfit, such as a picture or a text input field, can be done with the same user interface, and the finished product is always ready for testing by the developer or user.

2.8 Connectable systems and data sources

The main task of low-code solutions is to collect information from different sources and display them in one application indoors. Such solutions include, for example, the desktops of the principal user, where as the principal user miva person is able to see, for example, the different parts of the status of the service he manages from several different perspectives from the source. The benefit of this kind of solution is the increased availability of information and the a distraction when management can only be done through one application. Connection options vary vary by development platform, but as a rule contain the most common data formats such as the deepest databases, Excel tables and the most commonly used data sources.

Microsoft has released its still-in-development database solution Dataverse, which corresponds to the for the need to securely store data when using the PowerApps development platform. Dataverse works as the Microsoft product family's own low-code database, and utilizes cloud storage in data retention. It is also possible to regularly synchronize information to Dataverse from other systems and thus utilize one platform as the source of all information. (What is Microsoft Dataverse 2022.)

2.9 Shadow IT - problem

One known problem in the IT sector is the so-called shadow IT. This describes the phenomenon of where to work tools such as software are used that are not managed by the company's own IT, or whose use the company is not known at a general level at all. (Person 2022.) For example, if the company's kijÿ creates a program for the customer on a platform that no one else in the company knows about, or if the work the author uses a tool for project management that has not been widely used in the company.

The reason for these situations can most likely be found in the company's mistake in recognizing the need for employees for a certain software, or the employees' decision not to tell about their need.

Programs like this can bring problems when the employee changes jobs or be large security risk, enabling access through the program even to the entire company's internal network the size. Low-code solutions try to answer this problem, by offering tools with which no-do simple, time-consuming, easily repeatable tasks. With produced solutions we can automate these tasks, and free up resources for more important work tasks, and thus also promoted employee well-being at work. (Person 2022.)

2.10 Benefits

In software development, the first version of the program is initially planned, which the development begins to aim at-terrestrial. The term MVP (Minimum Viable Product) can also be used for this, i.e. directly in Finnish as the smallest profitable product. (Citizen Development - The Handbook for Creators and Changemakers 2022.) This aims to get a product produced as quickly as possible that the end user get to test, and so that any things noticeable during the test run can be fixed or modified based on feedback.

Low-code offers a fast route to this, as the product under development can be continuously tested in the development on the platform, even if not all functionalities have been implemented in the product yet. Compared to to standard development, the speed from production to finished product increases with a low-code solution hui-masti (see figure 3). As a result, the subscriber of the application also gets a functional product en-tice faster.

Speed plays a big role in the core of low-code solutions. According to a survey conducted in 2021 (see figure 3) the speed of low-code development was clearly faster in the opinion of the average respondent compared to traditional software development. If, for example, intended for internal use the software can be produced up to half as fast, which means directly saved working time and thus also saving for the company implementing the product. If the resulting product works as defined by the requirements set for it, it is clear why low-code can be the first to solve as an alternative when planning this kind of application. This enables, for example, software precise customized internal tools for your company and a sample product for a quick opportunity for presentation to the customer.

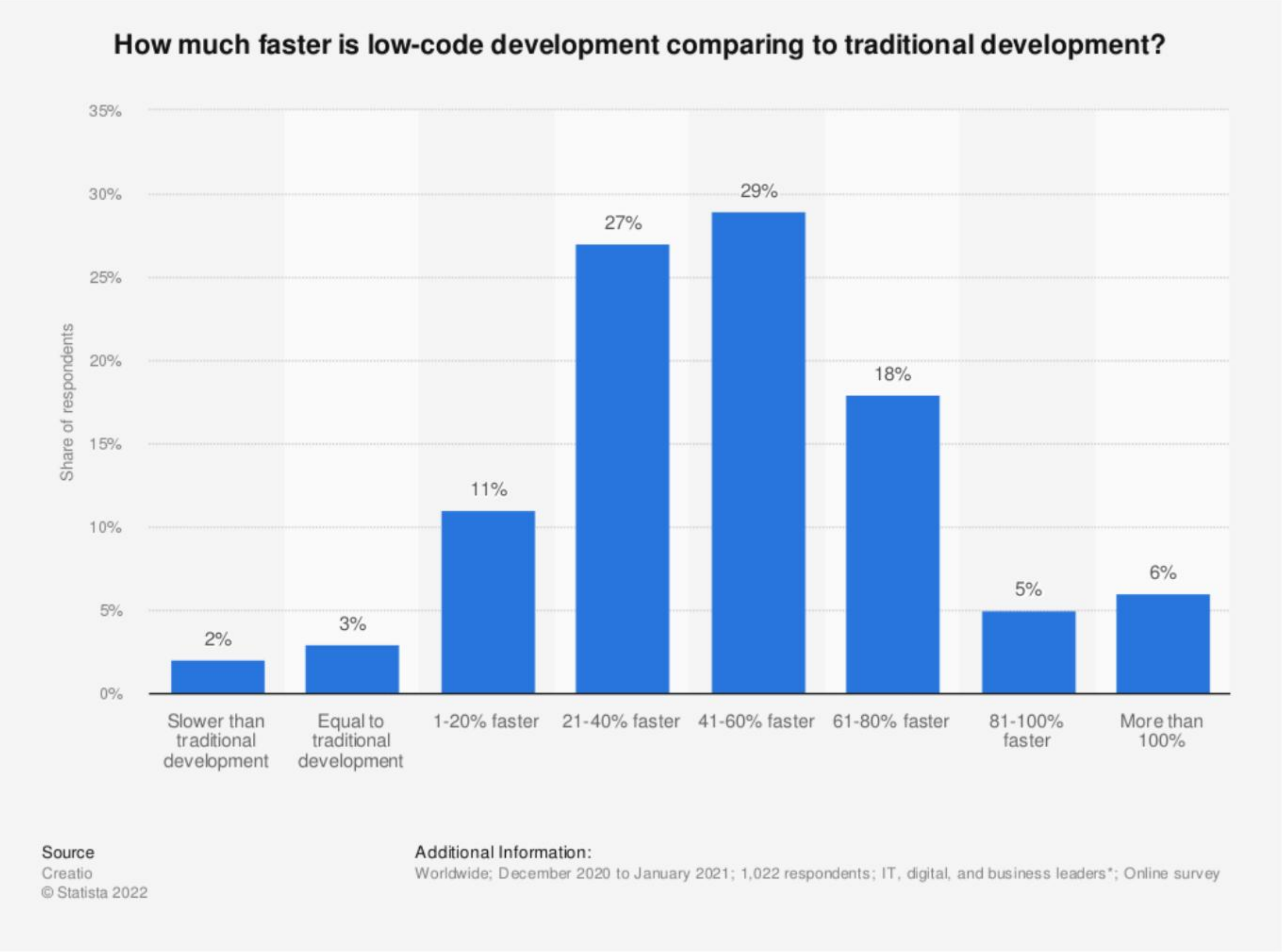


Figure 3. Perceived speed of low-code development compared to regular development (Vailshery 2022)

The resulting cost savings are in the interests of both the producing company and the customer. Low ones due to the adaptation possibilities of costs and development accelerating, the costs of production remain low and the development itself agile. Low-code products can be easily scaled precisely thanks to the implemented backend code generators. For example, processed data or user as the amount of software increases, in normal software development, one can end up optimizing the code in order to maintain the effective operation of the application, which takes time, generates more costs and in the worst case, even momentarily, can critically hinder the operations of the client company.

Because the platforms create the necessary back-end code at the moment when the user sets the component ke- in the hity platform's user interface, the implemented program must be constantly ready for testing, and there is no need for it for example, compile separately for testing new features. Thanks to this, the program's the hitter can constantly test the program, and make the necessary changes immediately after a new feature after adding. This also makes it possible to even engage the customer in the development rather, because the current state of the program under development can be constantly seen and tested. In this way, possible program change needs can also be reacted to immediately, and these avoid the high need for editing the program code, similar to normal software development.

Low-code platforms enable many of the same things that standard software development offers, although the programs produced with these are subject to the limitations of the platform. Earlier experience for an experienced software developer, low-code platforms offer good implementation opportunities to execute software even faster and more efficiently. By editing the operating code, develop- depending on the platform, even very complex programs capable of the same can be achieved a project of a larger size could be realized with a much slower schedule.

2.11 Disadvantages and limitations

Low-code platforms, like any other options, eventually bring limitations or disadvantages for development, the finished product, or its maintenance. When choosing a platform, you should take taking into account, for example, which service provider ecosystem is willing to commit to, what possible restrictions on the platform and for which other applications or services to produce the program needs connections. The background codes created by the platforms, which control the operation of the program, are often not compatible with each other, e.g. the status of changing the development platform in the

Since development is most commonly limited to one development platform, this brings that one as a limitation options produced by the platform provider. For example, it is not possible to add additional self-implemented components, due to which the software needs precise customization they can get caught in these. For this reason, the developer using the platform is the one producing the development platform limited to the company's innovation and solutions. In these situations, the question arises, will do I necessarily guarantee a solution that exactly meets the needs, or can we be satisfied if possible to a small difference in the desired function or user interface. When choosing a platform, it is it is useful to find out to which other systems, databases or data sources the platform in question able to combine and compare these with the needs of the program being produced.

Data security, as one area, has steadily become a bigger factor in the production of software here we go. Especially when handling sensitive information, the software developer must be aware of the possibility about additional vulnerabilities or data leakage risks. When choosing a low-code platform, we will sa- by choice, also the architect of the application's information security, the background code of ready-made applications being the code produced by the development platform. For example, if a function generated in the background does not execute correctly validating the user's input, the developer of the finished application is not necessarily aware of it of the risk factor without more detailed testing. When choosing a development platform, the developer presumably trusts, that the generator of the background code of the development environment is designed to function sufficiently for information security additionally to guarantee the data security of the program he created.

A low disadvantage in low-code software development is, due to the modifiability of the code, also the possibility possibility of the program breaking. If, for example, there is a change to the background code of the development platform in the section where the developer has added his own code. However, this is a low-level disadvantage, because the platform is usually informed about the content of future changes and possible effects on the products using developers. Due to this, it is possible to implement the necessary modifications quickly, without a larger investigation of the problem.

The lack of features or their limitation when locked to a specific development platform to the producer is one of the biggest disadvantages of low-code development. The answer to this problem for example, Microsoft collects a collection of user ideas with a forum-type solution of new features or modifications. Here users are able to vote on different ideas and the most requested of these will eventually be realized and implemented as part of PowerApps features.

3 Research layout

3.1 Research object

The low-code platform solution PowerApps produced by Microsoft was selected as the target platform for the research. The first public announcement of this happened on Microsoft's blog in 2015, where it says-

the solution of a company that is

to the problems of the number of developers of music apps (Staples 2015). It was first published in

kise for trial use in April 2016 (Desai 2016) and finally the official release took place

in October of the same year (Phillips 2016). PowerApps applications are indeed implemented mobile en-

sin - based on the principle, which first takes into account the view of mobile users when using the application. The solutions produced in this way can also be integrated into Microsoft's own ecosystem, e.g.

into a SharePoint page or as a separate Teams application. PowerApps was selected for the study

as one of the dominant producers of low-code platforms (see figure 4), and the Microsoft brand

for the sake of familiarity.

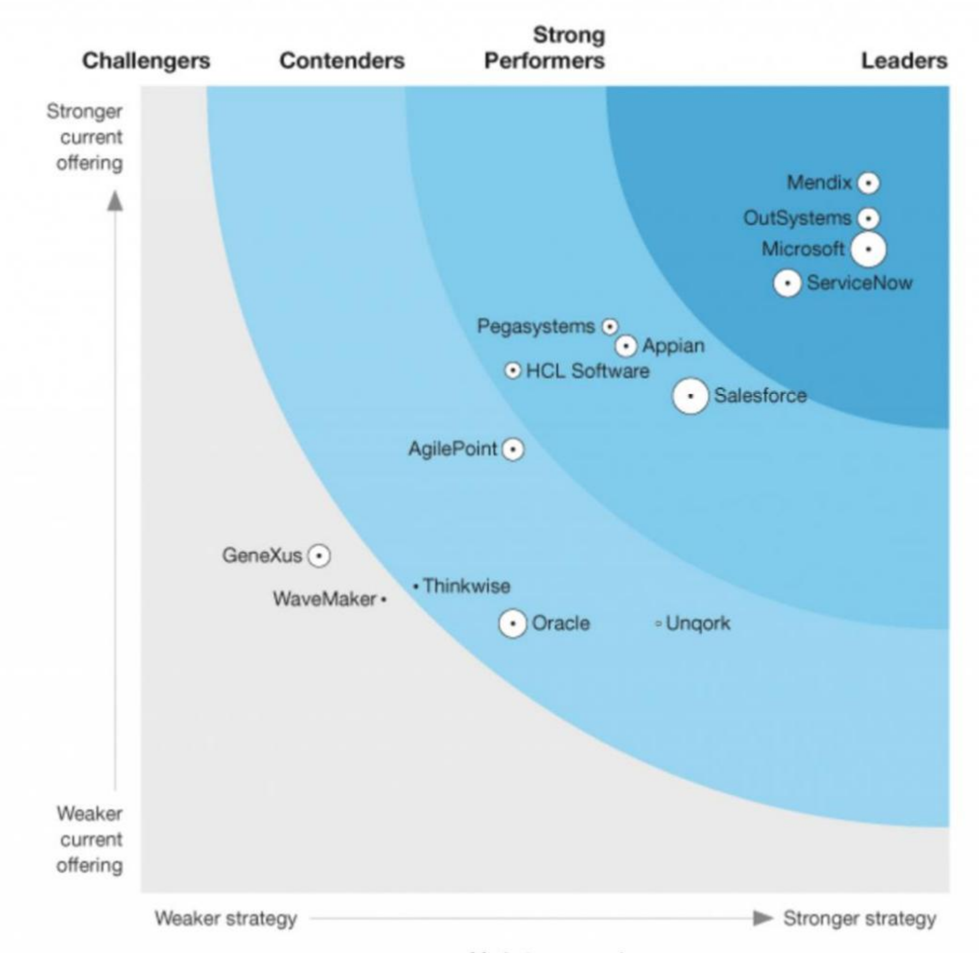


Figure 4. State of supply and strategy of different Low-code development platforms (Ramel 2021)

PowerApps is based on environment-based use, where the user can create independent additional environments for which different application solutions are produced, so that applications in different environments devices can use, for example, only the connections defined for that environment. This possibility lists the easier management of larger entities and limits the necessary connections and the use of boards according to the needs of the environment.

PowerApps is inspired by the PowerFX programming language used in the editable code The functions used by the Excel program and the commands intended for data management. Microsoft says that this is precisely intended to help the transition, especially for those with a business background for users who don't necessarily have more programming experience than different Excel spreadsheets in terms of functions. For its part, this facilitates the readiness of the group in question to implement various applica-implementations, offering familiar formulas to filter, organize or calculate information. If answer-but the command that executes the function was not found for some function, a model was taken for it as follows from the general data processing language, i.e. the SQL database language. (Microsoft Power Fxin overview 2022.)

3.2 Goal description

The aim of the study was to examine the process in low-code software development as a whole as well as to investigate the reasons for its growing popularity on a practical level. While doing the research, we observed also aspects related to the development itself, using Microsoft's PowerApps as the chosen platform development platform. When developing the application, we tried to take into account the advantages mentioned in the database as well as disadvantages, and looked even more closely at the creation of an application based on a low-code solution nin features, compared to standard software development. Low-code development and especially a lot of content has been produced about its future, on the basis of which research can be set basic assumptions, which are contrasted with what was experienced in the production of the example application.

Assumptions of the study:

1. Low-code software development is faster than standard software development
2. It is easy to implement a functional program on a low-code platform
3. Low-code enables a low threshold for route programming
4. A product produced on a low-code platform is just as functional as a product of ordinary software development

In order to obtain an experience base that can be compared to these assumptions through practical activity, the defini- the areas of software development to be taken into account and monitored in the production of the county's program with the following questions.

Production follow-up questions:

1. How quickly can you start development?
2. What was the experience of producing the application?
3. What were the biggest delays in the development progress?
4. What were the biggest constraints in the development?
5. What perceived benefits emerged in the development?
6. Were there any challenges or problems while developing the application?
7. Did the current popularity and measurements correspond to what was experienced in the development of the program?
8. Were the planned functionalities achieved?

3.3 Plan

A project management tool was chosen as the application to be implemented, which enables project-related management of tasks, distribution to project members and filling in general project information and monitoring from a common application for all project members. The application was made for an imaginary one for the company. More detailed information about the application can be found in the specification of the application's requirements in Appendix 1. The reason for choosing the purpose of use of the application was to produce a practical example that could implement in any company, and at the same time produce the most appropriate information from different areas of development.

Because the research took place, and the results of the research are derived from one person's experience pe- with rust, the results of the study are only indicative. When evaluating the results, it should be taken into account I believe that when carrying out a similar study, the results obtained may vary considerably depending on the possible different learning styles of the wearer of the research, mutual issues internalizing and applying previously learned in practice. The opposition of the research Luna are strongly influenced by publicly available surveys and the popularity around the topic and I make the starting points and the default values derived from them.

3.4 Research limitation

The comparison of different development platforms was excluded from the study, due to the characteristics of the comparison objects support. Each development platform is very broad in itself, often covering different functions, and different implementation methods for software solutions. A more detailed familiarization with different development platforms would require in-depth more familiarity with these, and this would divert resources away from investigating the development of the application itself and about the main topic. The development of the application itself takes place mainly in the same way on each platform, so the added value for research would be small compared to the amount of research.

The study does not go into more detail about the different functions of the development platform or the underlying system to the country that creates the necessary program code, due to the dimension of the subject areas in question. Different development in addition to similar features, the platforms also have different platform-specific functions and own background functionality for code generation. These are considerable in themselves, and to these the operation of the platforms based on them, which is why access to these is also blocked mainly in others on platforms except open source platforms. As the research focuses on user experience for the use of a certain development platform, demarcation is necessary in the absence of added value brought by the area.

The perspective of the research results is limited to those who have already gained some experience in the software industry. perspective, and does not take into account other perspectives, such as those of the inexperienced or more experienced possible experiences in developing the program. A more detailed user experience perspective man for review, a sample of user experiences from different levels of people, and research would be needed would be challenging to measure due to the variables of different learning styles and abilities, and not this one would bring added value to the research, which mainly focuses on the development work carried out on the platform itself.

4 Results

4.1 Start of the study

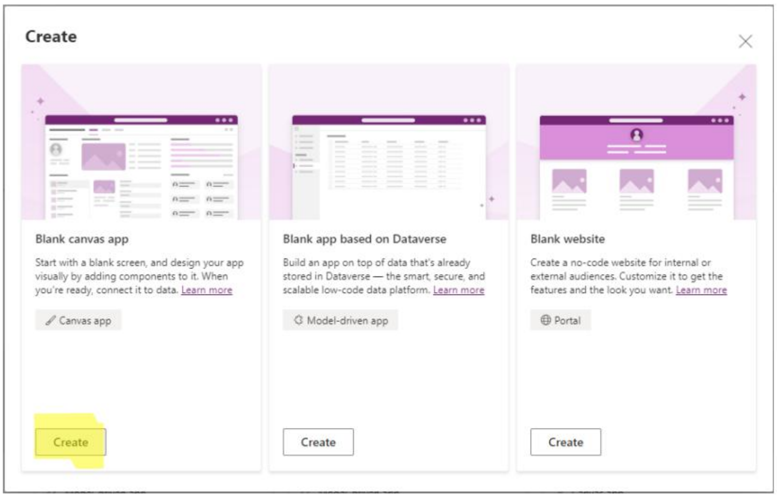
Initially, a narrow definition of requirements and purpose was created for the application (see appendix 1). For this descrip- ti's target group, purpose and targeted requirements for different functionalities. The actual program man's development started by creating a database based on a pre-planned database description, Using Microsoft's Dataverse.

Tables designed for the program were created in Dataverse, and relationships were added between the data it for connections. The application was defined to retrieve from the Users table information about users who have required license conditions for using PowerApps. This made it possible to use the application, while the user is logged in to his Microsoft account with a browser. This was not necessary for the application create separate user IDs. At the same time, also added to the user database of the fictitious company a new user would be directly ready to be added to projects, as long as the user's licenses are in order.

The application was started to be implemented with the least possible consequence of the instructions, possible I guarantee the most authentic experience of how quickly a new application can be implemented to do without previous experience.

4.2 Creation of the Application

The application was created by choosing an empty canvas-type application as the basis of the application. This creates a new of the application, without initial foundations or preset data connections to the application.



Blank canvas app

Start with a blank screen, and design your app visually by adding components to it. When you're ready, connect it to data. Learn more

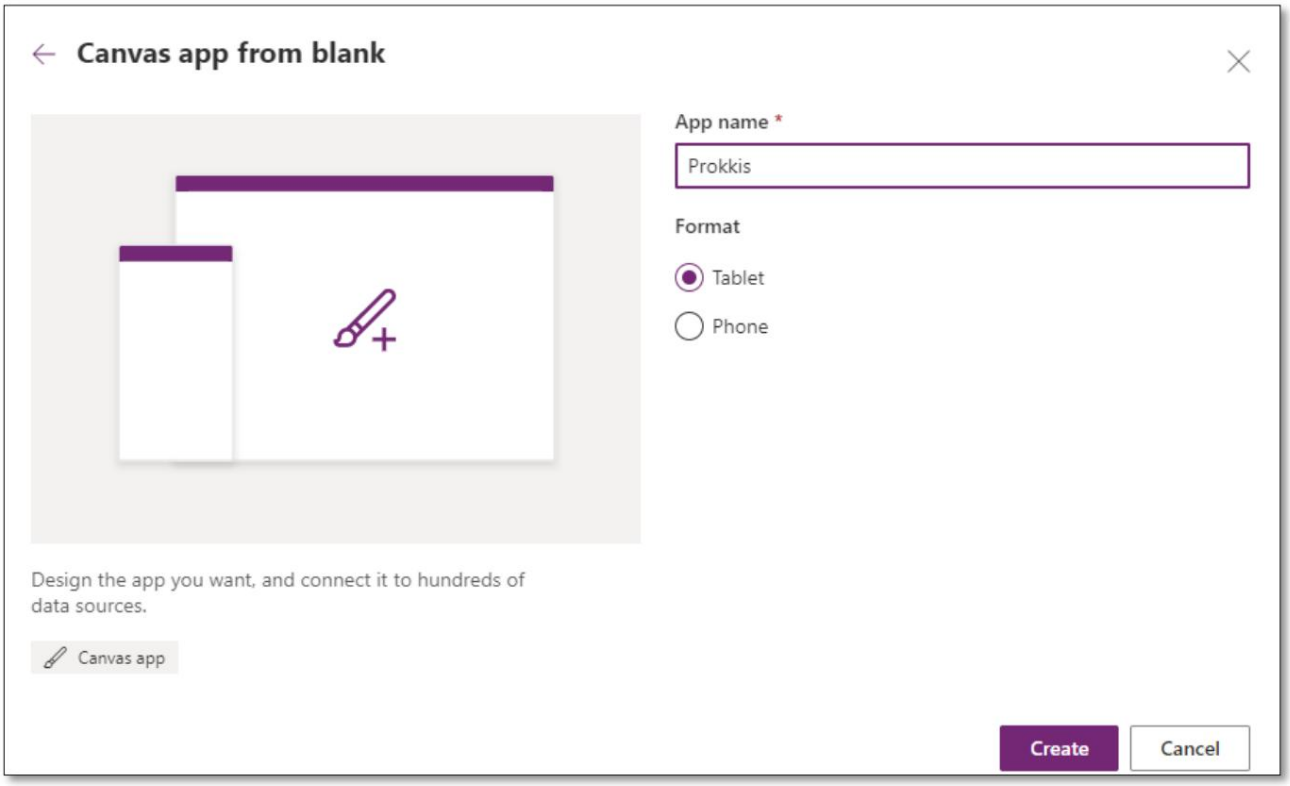
Blank app based on Dataverse

Build an app on top of data that's already stored in Dataverse — the smart, secure, and scalable low-code data platform. Learn more

Blank website

Create a no-code website for internal or external audiences. Customize it to get the features and the look you want. Learn more

Figure 5. New application options



Canvas app from blank

App name *

Prokkis

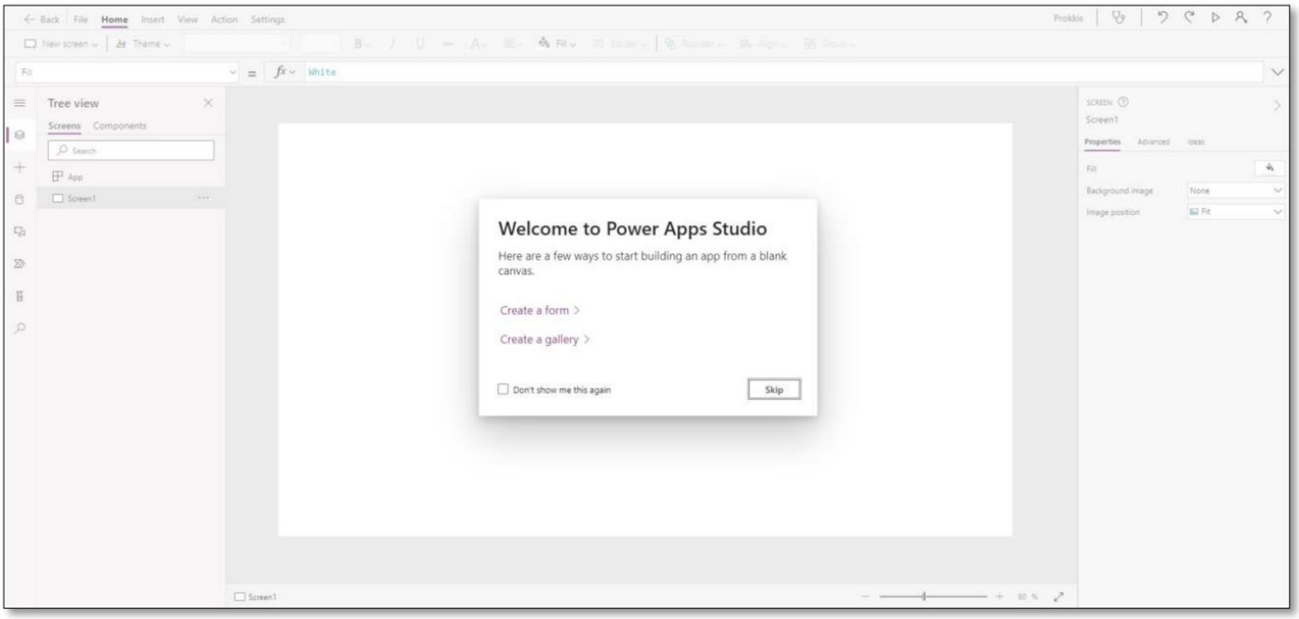
Format

● Tablet

○ Phone

Design the app you want, and connect it to hundreds of data sources.

Figure 6. Start options for creating an empty application



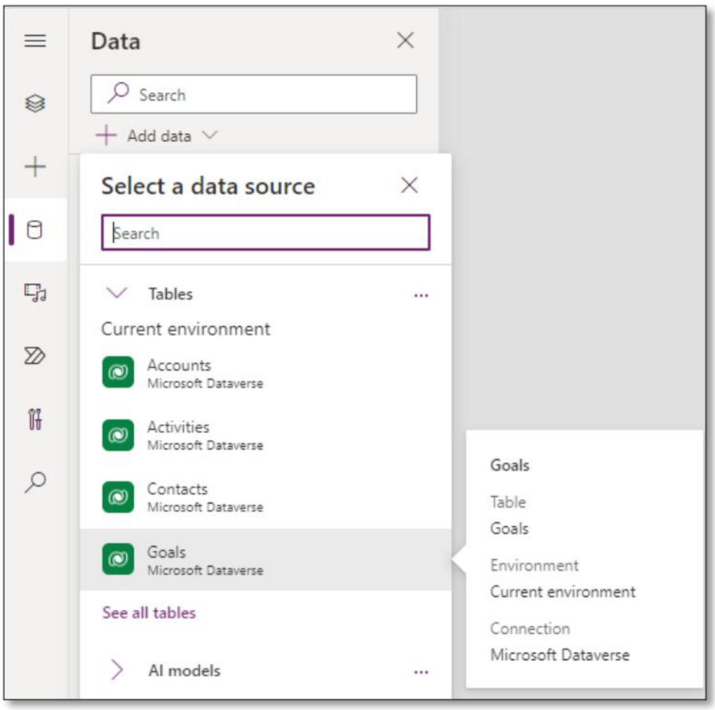
Welcome to Power Apps Studio

Here are a few ways to start building an app from a blank canvas.

Figure 7. Start view of the new application

4.3 Combining data with the application

Tables from the previously created Dataverse database were connected to the application by adding a connection to each to that board for the application to use. In this way, the information retrieved from the tables could be combined to create components and display this information to the user. A platform was also added to the application newly connected to Azure Active Directory, which always manages the users of that organization, but this was later found to be unnecessary, as the standard User table in Dataverse always searched the licensed users of that organization automatically.



Goals

Table
Goals

Environment
Current environment

Connection
Microsoft Dataverse

Figure 8. Combining data sources

4.4 Adding Components

The design of the start page was started by adding a new text component, displaying the name of the app-for what.

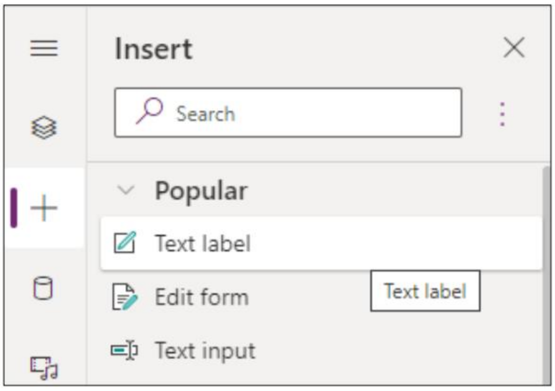
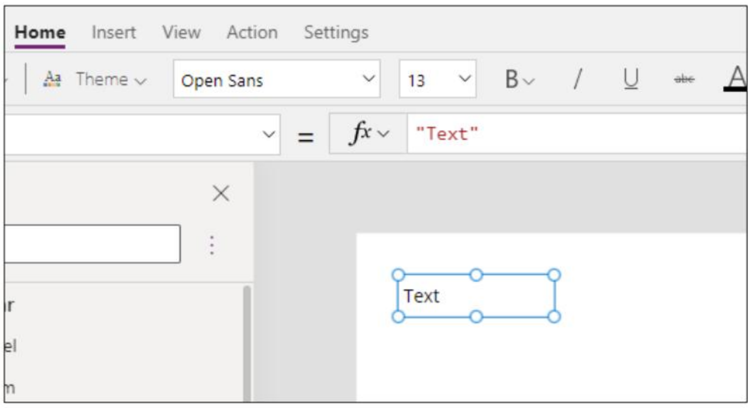


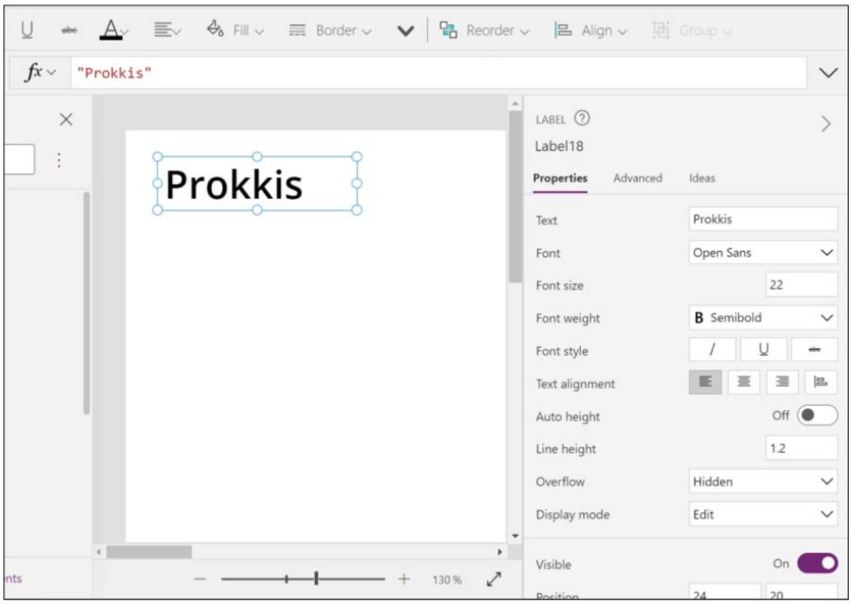
Figure 9. Adding a new text component



fx "Text"

Figure 10. Added text component on screen

The text was changed to the name of the application, and placed in the right place. The various properties of the component features such as font, size or color could be adjusted from the component's settings panel.



fx "Prokkis"

Label18

Text Prokkis
Font Open Sans
Font size 22
Font weight B Semibold
Font style / U
Text alignment
Auto height Off
Line height 1.2
Overflow Hidden
Display mode Edit
Visible On
Position 24 20

Figure 11. Detailed settings of the component

Different components were used to display different information also with the help of functions. Having been logged in for example, user information could be retrieved with the functions User().Image and User().FullName. These combining using an image and text component, a composite component was created (see fig 12) which displayed the user's profile picture and name.

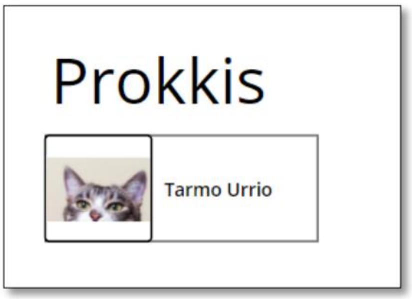


Figure 12. Display of user information with several components

4.5 Addition of displays

Screens were added to the application (see Figure 13) for different features, for example a project or a task for more detailed review and management. Switching between screens was done by pressing the through who and different transitions such as the creation of a new task.

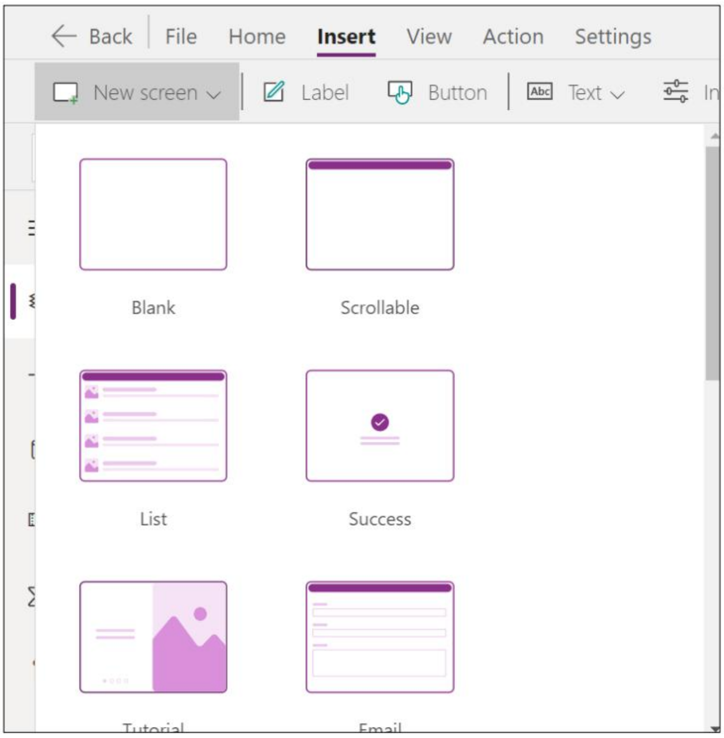


Figure 13. Adding a new screen

4.6 Management of displays and components

The application's sections were managed through the Tree view panel (see Figure 14), which in the listing all the different screens and components of the application were visible.

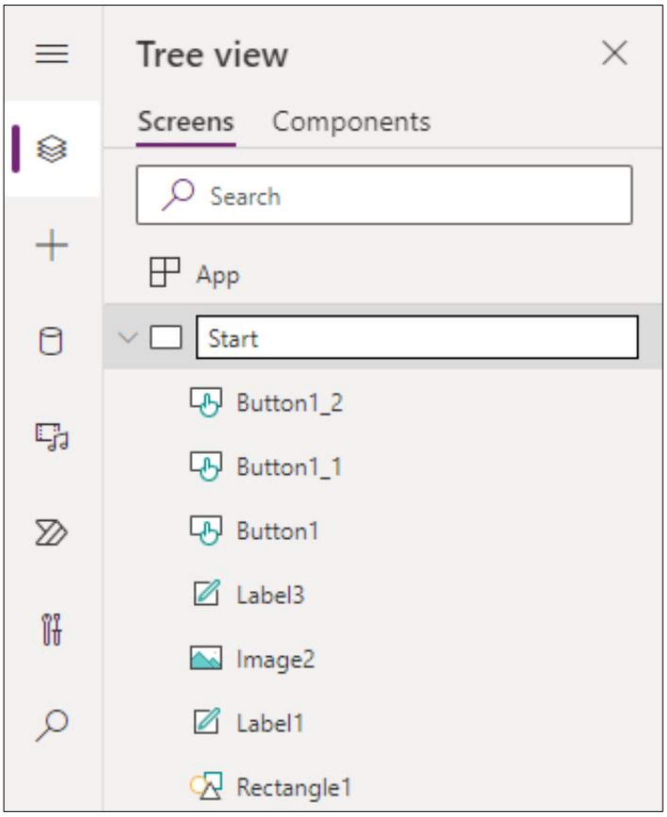


Figure 14. Tree view view

4.7 Miscellaneous Components

The information obtained from the projects was used with a ready-made diagram component (see figure 15). These li- was set in the project goals section to visualize the number of remaining tasks on the project, and to give an illustrative picture of the current state of the project's task load. Information to search was filtered to be project-specific and the information sections of the pie chart were defined to appear ready share of existing tasks.

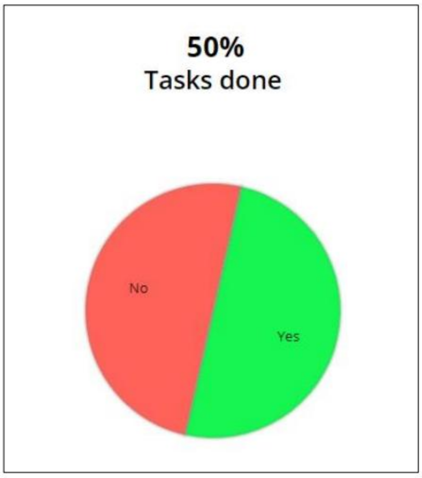


Figure 15. Information visualization

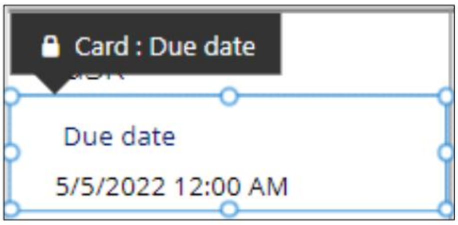
Ready-made data cards were used to display the different fields of filtered data for individual targets.

in the information sections. The data cards were defined as the right sources of data and the need for this data.

useful filters were added to the information search functions. Information cards (see figure 16) and gallery-type

the components (see Figure 17) automatically fetched and displayed the defined data from the data source,

gave good definition possibilities for even more complex delineation of the necessary information.

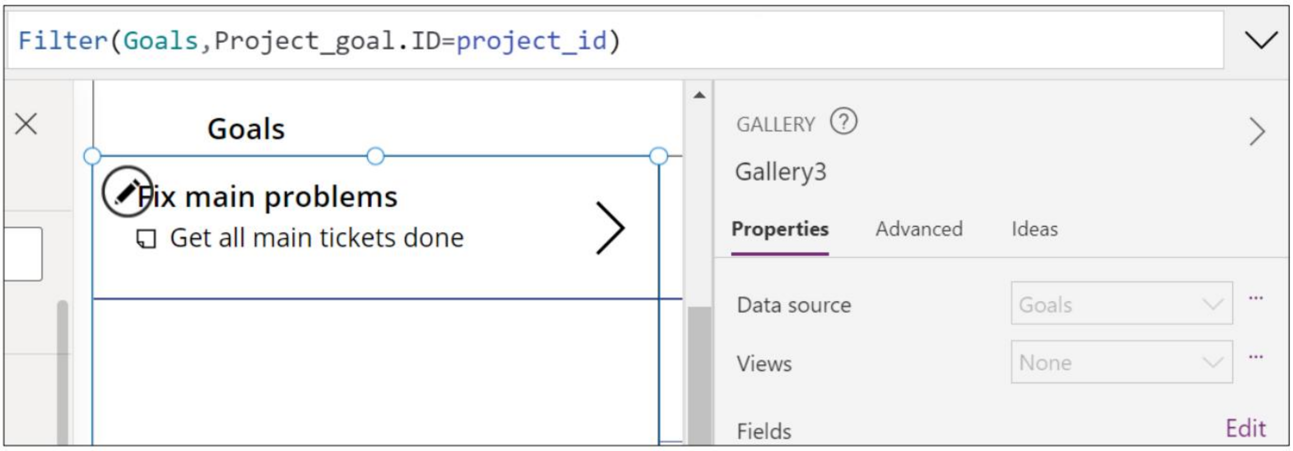


Card : Due date

Due date

5/5/2022 12:00 AM

Figure 16. Information card - component



```
Filter(Goals,Project_goal.ID=project_id)
```

Goals

Fix main problems

Get all main tickets done

GALLERY

Gallery3

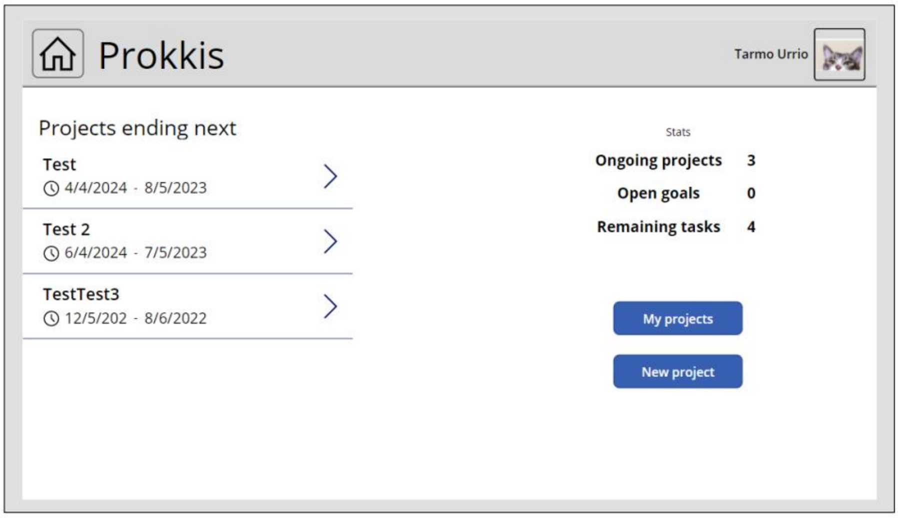
Data source: Goals

Views: None

Fields

Figure 17. Gallery – component

After connecting the data sources and adding more components, the application's home page al- came into being.



Projects ending next

Test	4/4/2024 - 8/5/2023
Test 2	6/4/2024 - 7/5/2023
TestTest3	12/5/202 - 8/6/2022

Stats

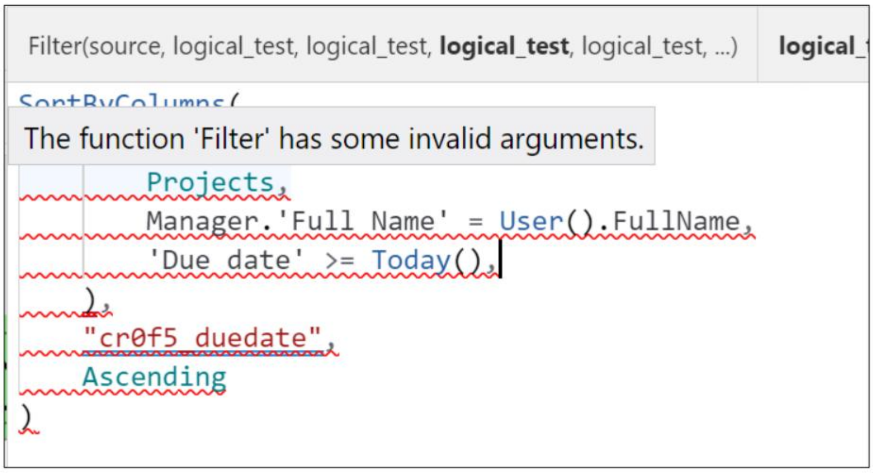
Ongoing projects	3
Open goals	0
Remaining tasks	4

Figure 18. Preliminary front page view

4.8 Error situations

If the application platform detected an error in the function used in data retrieval, the notification was clearly displayed in the editing section, underlined in red, and by telling in which part of the function the error is

come down



Filter(source, logical_test, logical_test, **logical_test**, logical_test, ...) **logical_**

The function 'Filter' has some invalid arguments.

```
SortByColumns(
        Projects,
        Manager.'Full Name' = User().FullName,
        'Due date' >= Today(),
    ),
    "cr0f5_duedate",
    Ascending
)
```

Figure 19. Function error message

On the side of the application view, the corresponding error also appeared clearly, illustrating the error and attention-machines (see figure 20), indicating precisely in which section the fault is located due to the error.

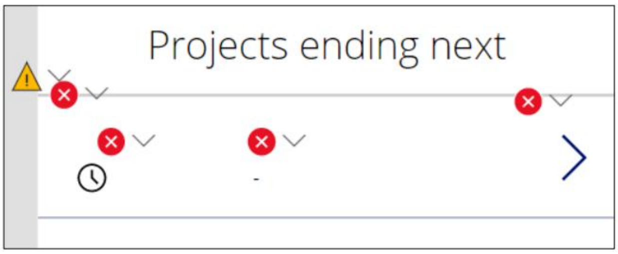


Figure 20. Error messages on the application view side

4.9 Defining own components

When completing the finishing touches, the same com-

ponents such as top navigation. Although the components were always copied from the previously worked screen due to small movements and modifications, the buttons were not directly

at those points, and thus when moving from one screen to another, you could distinguish their movement. Over here own components were found as a solution (see figure 21). The components can be any combinations of different parts created by the user, and function as easily added and copied combinations ti-

on the hips, where you want to repeat some functionality several times. Using these, you implement ti's top navigation and its functionalities on each screen.

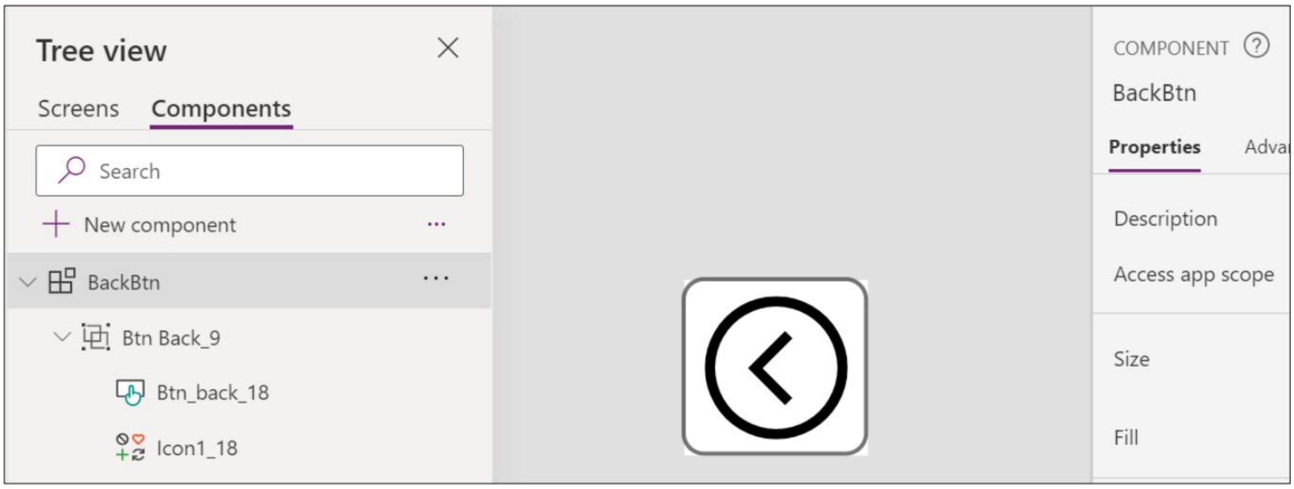


Figure 21. Creating own components

4.10 Application development

The implementation of the application progressed one screen at a time, at the same time testing functions and the display of data ac-

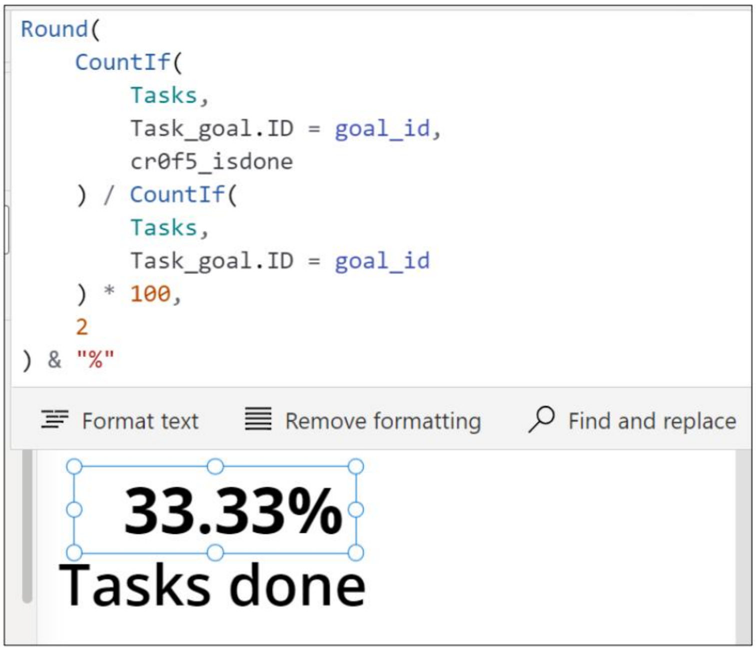
closely. After the initial version of the new screen was completed, the completed section was tested transitions and data transfer between different views. This is how possible points of error could be noticed-country and repair fresh when implementing the section. When making the application, attention was paid to so-to the usability and straightforwardness of vellus, so that the application could be used without instructions den chapter or more extensive familiarization. Because of this, we tried to keep the application as small as possible easy to read, using large enough font sizes, using clear selection buttons and information presentation and using logical progression paths in the application.

In encountered problem areas, the solution was often quickly found using PowerApps users with teisöforum, where users of the platform discuss different solutions and possible problems what they face when making implementations. When doing the research, typical challenge points were examined follow-up questions set for the study in connection with the road, and these were compared in situations to the experience gained from the development experienced until then. Problems were resolved how quickly the problem could be solved using the instructions of the application platform itself, or how easily the implementation options for the item to be implemented by posti were clarified.

4.11 Research results

The implementation of the program was tentative at the beginning, as the development platform was still new and unknown. Himself to-however, I was able to get the hang of it very quickly, by testing different components and thinking the next step and how a typical function is implemented on that development platform.
When this was caught, and the overall picture began to take shape, the implementation itself was only the goal of the solution. depending on the selection and implementation, you could see why low-code solutions are research based on its initial assumptions, significantly faster to produce compared to the usual application to development. After the initial phase, these assumptions began to shape into obvious facts, while often thinking about the simplicity and intuitiveness of the implementation itself.

The biggest challenges in implementing the application were displaying the information from the database, pro-delineation of searchable user IDs to be selected for projects, as well as defining the selections of different forms lyt. Creating the database tables and the relationships between them was an effortless experience, and if the user has any previous experience in creating a database, this can be done without better familiarization with the documentation. A problem arose when filtering the information available here often a small amount of misrepresentation. The error message of the development platform said that the search for information failed essa, for example that some information cannot be filtered or you just underlined the part from the background code saying that that part of the function is wrong, but couldn't figure out exactly which the passage itself is incorrect. In the end, however, a solution was always found for these points, and the problem il- the confusion caused by the chaining of several functions often went away. Because PowerApps always gives one input field for the function of the component, it leads to a situation where different functions are chained but several nested, for example, when narrowing the information to be searched. When, for example, you want to search the database pin the results in which more than one condition is met and which checks the ones in question in several different areas of data, the function generated for this search field grows to the length of several lines (see figure 22).



```
Round(
    CountIf(
        Tasks,
        Task_goal.ID = goal_id,
        cr0f5_isdone
    ) / CountIf(
        Tasks,
        Task_goal.ID = goal_id
    ) * 100,
    2
) & "%"
```

Figure 22. Combination of several functions

Despite this, the creation of functions is well assisted, and the development platform provides instructions for each one at the point of function creation, from the definition of the information needed for that parameter in the development environment in a typical way for crossbreeds. This supports learning new things quickly, repeated guidance through, and the most commonly used functions were internalized relatively quickly.

As the implementation progressed, a recurring problem was seeing the code running in the background of the platform lack of. Since there was no direct view of the code, all information about the functions made and the choices selected of those remained behind the data windows. At some points, this slowed down the progress to unnecessary repetition as an operation, because recalling the value of a set variable or color, for example, was more behind switching between pressing man and on screen, then had to go back to the screen you are working on in the same way.

Low-code development platforms are largely limited environments, but still during the research no You obviously have more experience of being limited, in terms of platform limitations or opportunities. A few preliminary replacements were available for displaying searchable information and functionalities condition or from a ready-made alternative he was able to create his very own look. The appearance of the application is based on on the road and during implementation, the limitations of the limited appearance side of the platform appeared as a challenge. On the platform, the main options for editing the appearance were font, background and borders colorings. In addition, for example, as the only option for stylizing shapes or appearance, components you had rectangular blank graphic elements that couldn't be rotated. go-there was also an icon library produced by Microsoft, but these were also mainly tried as visual elements in connection with the information of different components, and not so much by themselves for the implementation of the layout.

Despite the paucity of the graphic side, the benefits of the development itself came out well in the development here. Once the appearance has been considered, and the initial feeling has been reached, for example the implementation of a new screen tus was a very streamlined and pleasant experience. Data merging and searches took place auto-logically, as long as you limit the incoming information as needed and the functionalities work directly after the addition of a new component without multiple iterations to ensure functionality for January. Continuous possibility to test the application by simply holding down the alt key it further accelerated the already fast pace of development and gave the opportunity to focus on the to produce vellum without repeatedly having to move between the development environment and between application perspectives, as for example in typical web application development, new compo-when testing nent.

However, the biggest difficulties in carrying out the research were not directed at Power Apps itself. values, but rather the complexes of Dataverse tables selected as a database solution for managing and connecting data structures with each other. For example, the database standard table User, provides information about all users under that organization who have the necessary licensing you can use the platform's solutions. One problem was several users at the same time nor adding to the project, because more than one user-type record could not be added to one to the data line of the board. This challenge was circumvented by simply giving the user an option only add one user to the project at a time, and using a dashboard that connects the to the project, because finding a better solution to the challenge in question would have guided the deviate from the research goal of the target development platform itself. This also contributed attention to the initial assumptions of the research about the supposed good parties of the development itself. big-in the absence of small challenges in the development itself, it can be stated based on the research that the development the advantages in speed and simplicity of development brought by friends Articles, opinions and public attitude in favor of low-code solutions increase this very high in the options of application solutions, and they seem to be correct. Development itself is very straightforward and if the implementer has any previous software development background, the only challenge in implementation is getting to know the environment and its operating style.

Although the application performs basic functions and is not complex in terms of interfaces or logic, producing the application in practice produced a lot of information about the process needed for this, different about them and the best practices that should be followed when creating a new application. Microsoft is also created a handbook for PowerApps development that covers best practices and standards, such as naming conventions for different components or views when creating an application. The manual in question however, jaa is not meant to be read necessarily before the start of application development, but rather, after the first touch of development, to shape the production of future applications by instructing the practices found to be the best. (Baginski & Dunn 2018.)

In the end result of the application, it was well suited for its purpose and the application was defined to specific requirements. The application worked in accordance with its intended purpose and works for its purpose as a simple tool for managing the project and its goals and tasks. Produced the goal of the solution's functionalities was to act more as test objects than as goals with which made it possible to familiarize yourself with the platform based on an example solution. These produced very experi-about the development itself and its different areas, offering a comprehensive view of the different development platforms possibilities and features. The areas of production of the necessary information, functionalities and from combining the layout work seamlessly together and producing the application was very a pleasant experience that encouraged me to continue learning about the subject by leaving a positive impression likuvan's application development on low-code platforms.

The experiences gained from the research strongly support other research results on the subject, ol-less very sympathetic with the initial assumptions set for the research. The speed of development, which got the application to the first working version and the opportunity for continuous testing when the various functions of the application are completed, well support the claims about the advantages brought by the development model both for the application producer and the client. Based on the results, it is possible to assume fairly safe masti, that low-code development will become much more common in the next few years, especially as the need for applications produced for the internal needs of companies increases, the digitization of companies attached.

Even though the development work itself was relatively easy and simple based on experience, it should be taken into account that experience is greatly influenced by previous experience and know-how in software about the development of For a citizen developer who does not necessarily have experience creating an application est, or bases by the most commonly used functions, you can catch up on the development- which will definitely require more familiarization. However, the platform itself offers the usual compared to software development, a very easy route to developing an application, simpli- by many general aspects of the operation of applications, such as processing the necessary data or different functionality of sections. However, with comprehensive documentation and an active community, PowerApps is a good destination for starting low-code development, offering good resources on the subject for learning and solving problem situations.

5 Reflection

At the beginning of this work, I didn't know much about low-code development. I based my information on what I heard and what I saw from the side of studies and working life. Without previous experience, I based my first my impression of the development is very much related to the speech and what I saw. Taking into account the development the constant growth in popularity of the habit, not hearing about the subject has been almost impossible studying what field. Often, as the popularity of a subject area increases, a constant in the long term is its popularity also deserved due to the characteristics of the subject. This often raises questions in the listener whether the destination offers what it promises. The first questions that surfaced for me were which ones are the possibilities of the product produced with low-code development and how flexible the development itself is.

The general opinions used as default values were very positive, and the promises of speed about development and comprehensive possibilities, did their part to increase interest in that matter towards the subject area. If something is so good, why not implement a large part of it this way? Subject area also aroused a certain amount of skepticism in my mind, because somehow the program is produced in such a way that control

is not fully himself, produces the experience of a shortcut to the end result and of uncertainty towards the final quality of the product. In part, this experience can be based on threat images of one's own know-about its necessary reorientation as the development trend continues longer, or because I am during my studies, I have adopted the processes of the usual software development method. The reason for the subject area's produc-fear of my own perceived lack of competence and getting used to the new
i do

As the research progressed, my opinion on the subject changed considerably, as the product implementation itself in practice, ordinary software development, but only in a significantly simplified and accelerated soon. With my biggest problems focusing only on displaying information, I often found myself pondering vani how effortless it is to implement the idea itself with the help of the platform. Graphical user interface significantly corresponded to the operation of tools intended for layout design or for example development of user-interface programs in the C# programming language, and I have some experience in these old, a similar graphical user interface was very straightforward to adopt.

Low-code development has been common knowledge for a long time, and although this has been predicted revolutionizing software development for several years, it feels like the real breakthrough is here occurred in the last two years, most probably partly as a result of the global pandemic that. As a large part of work moved online, the need for employers to have more software that supports business operations, because when employees are online, it is also necessary at work tools to be new application solutions that support work. Low-code for these needs the development corresponds perfectly. When the development of the necessary software does not necessarily require a large implementation and price and quality meet the needs, the implementation of the solution with low-code can be almost self-clarity for the company. Based on my own experiences, these are very likely selling points teja, and the threshold for ordering a program or even producing it yourself can be considerably lowered by these with benefits. Due to this, both the supplier and the customer benefit from the development method, the solution great benefits for both.

Low-code development offers exactly what it was designed for. My own knowledge of development was al-linguistic level before the start of the research, and I feel that I developed quickly during the research in the development work of that platform. The development is very complicated and many times when I was thinking about something I hit the rat-I get hooked surprisingly quickly just by trying it out, without much necessary research. Almost all the options in the different sections are streamlined and simplified, but still offer the possibility for a more detailed definition of the settings. This enables, for example, the developer to nicer toolkit, but at the same time also offers a less skilled user the ability to conditions for execution. Although low-code platforms are also targeted at people with a business background users, I felt that without experience in programming, or similar development work, it would be project could have been significantly more difficult to implement, because programming knowledge gives an efficient shortcut to development on those platforms. By knowing in advance already common functions and typical the fundamentals of the program's operations, the platform remains to be learned about the development of the platforms own and the language practices of the editable section of the used background code.

The biggest challenges encountered were related to searching, filtering and displaying data. This compact this is almost the basis of low-code itself, in which field it fulfills its purpose brilliantly. Companies the amount of data in use is constantly increasing, and this often results in a need or an idea for all this information utilization. For example, data obtained from projects, error situations or production can gives a comprehensive overview of the state of the company and ultimately enhances the employee's work when several asta, information from the source is combined into one and the same destination. Information management and analytics the challenges are mainly focused on the data analytics side, and for these purposes, develop their own applications, but for typical company needs, such as the for recording springs or, for example, replacing several main user control panels with one solution, has its own low-code to produce working solutions.

PowerApps as a platform is a great solution, especially for companies focused on the Microsoft ecosystem. for it. Easy connection to get information from, for example, a SharePoint list, increases in itself also the ability of applications produced with Microsoft's own solutions to offer versatile and agile solutions for different needs. Mainly, I would see the small and medium-sized business applications that speed up, for example, the workflow or process data in several from the source. The low-code implementation also enables product scalability and additional features producing mushrooms without major challenges, and thus also guarantees a good starting point for the further development of the product for development.

As the results of the study are based on the experiences of one person, the work of one development platform ly, the validity of these results must be highly critical. However, when comparing concrete advantages of the development method, it is possible to verify some areas with moderate certainty in the It can be stated that the time spent on product development is significantly reduced with low-code in the implementation of the platform for several reasons. When the implementation takes place online, the development environment itself there is no need to set up the work, and with the visual aspect constantly present, the user interface and view in project-based development, the current state of the product is constantly visible. this enables corrections, for example, in modifying the way the retrieved data is displayed, and when the data retrieval itself takes place immediately, it is also ensured that the code works at the moment it is written.

For example, when connecting the data display part to the desired database table, filtering the retrieved data when doing this, you can make sure that the search expression is working immediately, because in the event of an error, the development the platform gives immediate feedback on the inactivity of the written function. Also for example testing the functionality of the newly added button can be performed immediately by the functionality of the button after defining the The program created using the PowerApps platform can be tested even only by pi- by pressing only the Alt key at the bottom. Because the background code that maintains the program's functions is lessa is constantly up-to-date, it also enables the program to be continuously tested and ready for operation. In this way, the program developer can immediately verify the functionality of the section he has just created, without producing development a bigger interruption to work.

While carrying out the research work, I missed more visual elements in the application. Although in a large part of the components, the selection possibilities for different visual sections and edit- were good for the seasons, when implementing the development I was left with the feeling that the exterior looks very reduced, when the application uses similar components and information display styles. A big possibility the point is that I also missed the more precise visual editing possibilities - matte, because there were often many editing targets of different properties. My attention was focused often only to the component's fill color, and the effects of different user actions on this color. program appearance contributes to the user experience of the product, and even if the program performs the neces- common functionalities, the well-planned and coordinated appearance of the application brings added value to the application to its user.

After further reflection and testing, I found the solution to this was the self-produced image files, which- with the help of the necessary elements of different shapes could be added using image components.

For example, a circular visual element was obtained by creating the shape in an image processing program and removing the background coloring from the image. By combining this component with the standard one to the rectangular component (see figure 23), the generated program will have more opportunities to use for the planning and implementation of the work interface.

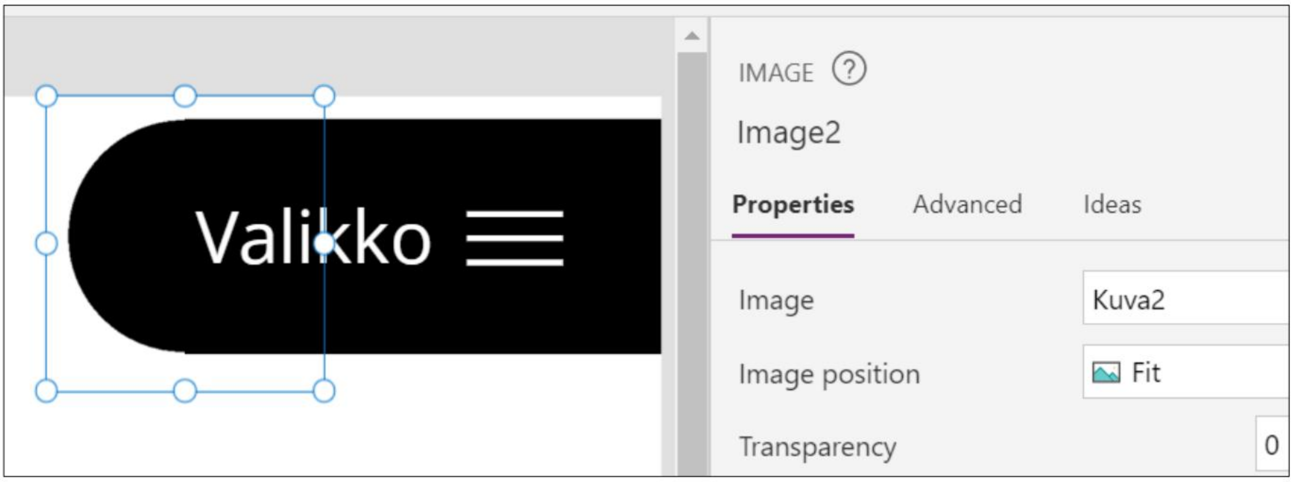


Image Kuva2

Image position Fit

Transparency 0

Figure 23. Using the image component visually

As a further development idea, it would be interesting to study different development platforms and their properties. Different sorts of there are a huge number of development platforms, and all of them have their own operating styles and differences compared to it. For example, the various connectivity options of the platforms enable an almost limitless r options for the implemented application and its functions. A closer look also pre- as a sign of the user experience and the comparability of products produced with low-code to ordinary ones software development products from the end users' point of view could be very interesting researchable, precisely with regard to the differences in the final products of the development methods.

Although the obtained results fully supported the default setting set at the beginning of the study, it is final- nen image of the development itself is always developer-specific. Individual opinions, attitudes and past experiences affect the experience of development and, like differences and preferences in different between jelming languages, low-code will certainly face a similar confrontation as well for its part. However, it can be said for sure that low-code development will become more common in the future over the years even more, and to grow into the most commonly used application solutions alongside the usual development style, bringing necessary new skills and new capacity for application to production.

Sources

Bock, AC & Frank, U. 2021. Low-Code Platform. Referred on 26/04/2022. https://doi.org/10.1007/s12599-021-00726-8.

Citizen Development - The Handbook for Creators and Changemakers. 2022. Handbook produced by the Project Management Institute. Referred on 26/04/2022. https://app.knovel.com/hot-link/toc/id:kpCDTHCC01/citizen-development-handbook/citizen-development-handbook.

Desai, D. 2016. Announcing Public Preview for PowerApps. Referred on 27/04/2022. https://powerapps.microsoft.com/en-us/blog/powerapps-public-preview.

Forsyth, A. 2021. Low-Code and No-Code: What's the Difference and When to Use Which?. Referred on 25/04/2022. https://www.outsystems.com/blog/posts/low-code-vs-no-code.

Garcia, K. 2021. What is low code? A comprehensive guide. Referred on 26/04/2022. https://retool.com/blog/what-is-low-code .

Lasar, M. 2019. 30-plus years of HyperCard, the missing link to the web. Referred on 05/27/2022. https://arstechnica.com/gadgets/2019/05/25-years-of-hypercard-the-missing-link-to-the-web.

Lindroos, K. & Björklund, S. 2021. Are there alternatives to the vilified Aster patient information system? Two domestic companies believe they can do the same more cheaply: "It can be done much cheaper". Referred on 23/04/2022. https://yle.fi/uutiset/3-12107472.

Lindström, S. 2021. Application development does not always require coding skills - A citizen developer's trump card is a personal understanding of business needs. Referred on 28/04/2022. https://www.itewiki.fi/blog/2021/06/applyoskekhetish-ei-aina-vaadi-koodaustaitoja-kansalais-kehittajan-valttikortti .

Microsoft Power Fx Overview. 2022. Article on the Microsoft documentation pages. Referred on May 2, 2022. https://docs.microsoft.com/fi-fi/power-platform/power-fx/overview.

Person, D. 2022. How Low-Code And No-Code Tools Can Help Stave Off Shadow IT And The Great Resignation. Referred on 26/04/2022. https://www.forbes.com/sites/forbestech-council/2022/02/09/how-low-code-and-no-code-tools-can-help-stave-off-shadow-it-and- the-great-resignation.

Phillips, J. 2016. Microsoft PowerApps and Flow are generally available starting tomorrow. Referred on 27/04/2022. https://blogs.microsoft.com/blog/2016/10/31/microsoft-powerapps-flow-generally-available-starting-tomorrow.

Ramel, D. 2021. Microsoft Power Apps Ranks High in Low-Code Research Report. Referred on 26/04/2022. https://visualstudiomagazine.com/articles/2021/05/13/forrester-lowcode.aspx.

Schoettle, A. 2021. No coding skills? Increasingly, that's no problem. Indianapolis Business Journal, vol. 42. Cited 2022-04-28. https://www-proquest-com.ezproxy.jamk.fi:2443/trade-jour-nals/no-coding-skills-increasingly-thats-problem/docview/2506760598/se-2.

Stamford, C. 2021. Gartner Says Cloud Will Be the Centerpiece of New Digital Experiences. Referred on 28/04/2022. https://www.gartner.com/en/newsroom/press-releases/2021-11-10-gartner-says-cloud-will-be-the-centerpiece-of-new-digital-experiences.

Staples, B. 2015. Introducing Microsoft PowerApps. Referred on 27/04/2022. https://blogs.microsoft.com/blog/2015/11/30/introducing-microsoft-powerapps.

The Low-Code Development Guide. Nd Presentation of low-code development on the Outsystems website. Dated 25 April 2022. https://www.outsystems.com/guide/low-code.

The state of low-code/no-code. 2021. Report on the Creation pages. Referred on 26/04/2022. https://www.creatio.com/page/sites/default/files/2021-05/Report-May.pdf.

Tozzi, C. 2021. Cited 4/5/2022 https://www.itprotoday.com/no-codelow-code/evolution-low-codeno-code-development.

Vailshery, L. 2022. How much faster is low-code development compared to traditional development?. 2022. Cited on 22/04/2022. https://www-statista-com.ezproxy.jamk.fi:2443/statistics/1254662/low-code-development-speed-compared-tradit-ional-it.

Vailshery, L. 2021. Low-code development platform market revenue worldwide from 2018 to 2025 (in billion US dollars). Referred on 22/04/2022. https://www-statista-com.ezproxy.jamk.fi:2443/statistics/1226179/low-code-development-platform-market-revenue-global.

What is Microsoft Dataverse. 2022. Article on the Microsoft documentation pages. Referred to 4/5/2022. https://docs.microsoft.com/en-us/power-apps/maker/data-platform/data-platform-intro.

Attachments

Appendix 1. Requirements definition of the example program

Requirements specification - Project management tool "Prokkis"

In brief

The company has identified the difficulty of project management as a challenge. Members of projects often do not have accurate understanding of the status of the project as the project progresses and, as a result, the timing of own tasks lutus can be challenging, or these have to be reviewed with the project leader. This was solved by a project management application will be implemented as a project to facilitate project management, monitoring and progress where from. The tool is implemented using Microsoft's low-code platform PowerApps. In the testing phase the application will be available to all persons participating in new projects. The market has used several different project management tools, but due to the small size of the company, it was decided to produce a project management tool internally, on a ready-made development platform as part of the becoming familiar with it and increasing the range of services.

The technologies used

ÿ PowerApps
ÿ Dataverse
ÿ Azure Active Directory

User groups

1. Project leader
   - o Manage project status
   - o Defines tasks for the project
   - o Assigns people to tasks
   - o Defines goals for the project
2. Project member
   - o Manages the tasks assigned to him
   - o Indicates the status of tasks

User stories

As a member of the USER001 project, I want to see the status of my projects from one view
As a member of the USER002 project, I want to see the tasks assigned to me
As a member of the USER003 project, I want to see the goals related to the tasks
USER004 As the leader of the project, I want to manage my projects
As the leader of the USER005 project, I want to assign tasks to users
USER006 -As the leader of the project, I want to set goals for the project

Characteristics

FEAT001 - Project management tools (add / edit / delete / status)

FEAT002 - Task management tools (add / edit / delete / status)

FEAT003 - Goal management tools (add / edit / delete / status)

FEAT004 - Managing project members (add / remove)

Outlook

VIEW001 - List of projects

VIEW002 - Project Status ÿ
Tasks ÿ Goals

VIEW003 - New project
VIEW004 - New task

VIEW005 - New objective

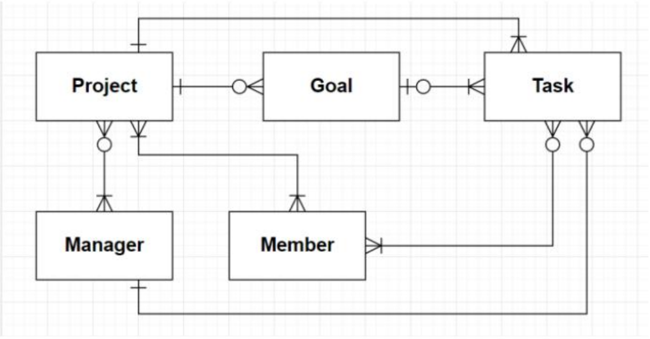
VIEW006 - Own projects
VIEW007 - My assignments

Database

Generated using Dataverse as described.

ÿ Project o ID
- int

o Name - string o
Start date - date o Due
date- date o [Project-
>Tasks] - Task o [Project->Goals]
- Goal o [Members] - Member o
[Managers] - Manager ÿ Goal

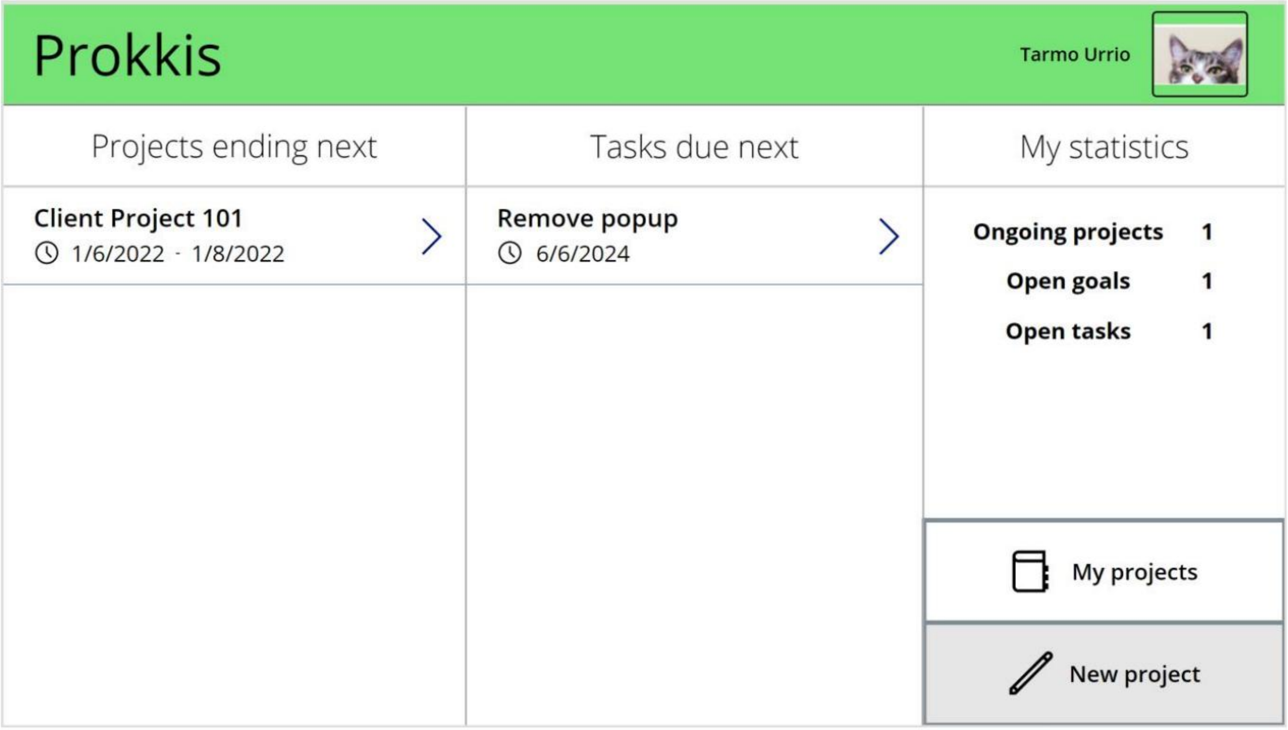


o ID - int o
Name - string o
Description - text o Due
date - date o [Tasks] -
Task ÿ Task

o ID - int
o Name - string o
Description - text o
Assigned to - Member/Manager o Due date
– date o [Goals] - Goal
ÿ Manager o Data
from Active
Directory
ÿ Member
o Data from Active Directory

Appendix 2. Layout of the finished program

Home page

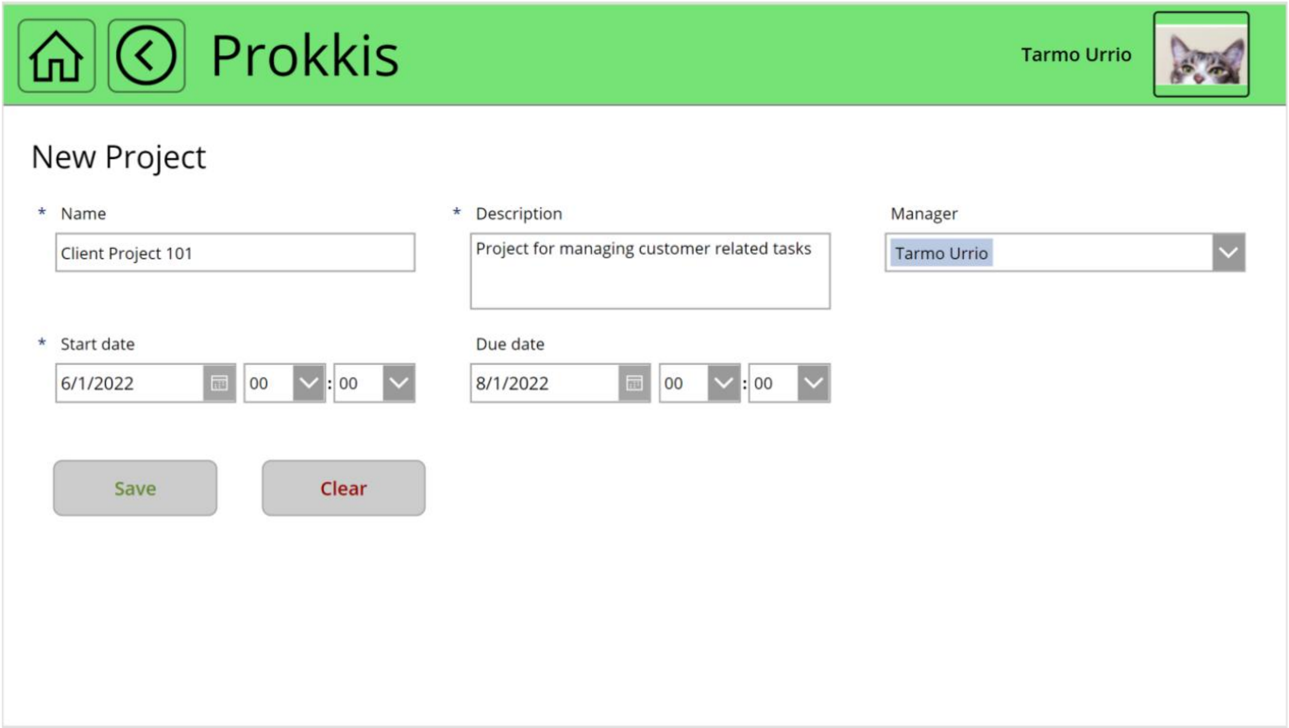


Prokkis

Tarmo Urrio

Projects ending next	Tasks due next	My statistics
Client Project 101 1/6/2022 - 1/8/2022	Remove popup 6/6/2024	Ongoing projects 1 Open goals 1 Open tasks 1

Creating a new project



Prokkis

Tarmo Urrio

New Project

* Name: Client Project 101

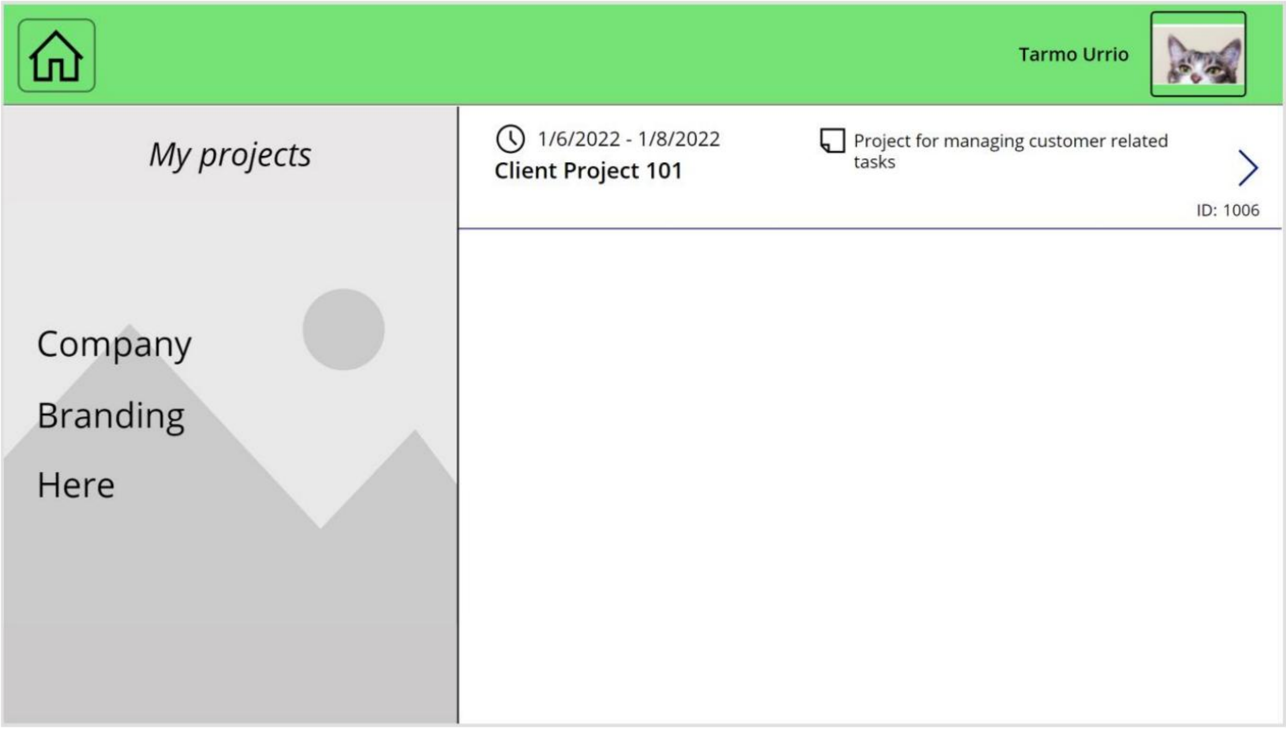
* Description: Project for managing customer related tasks

Manager: Tarmo Urrio

* Start date: 6/1/2022 00 : 00

Due date: 8/1/2022 00 : 00

My project

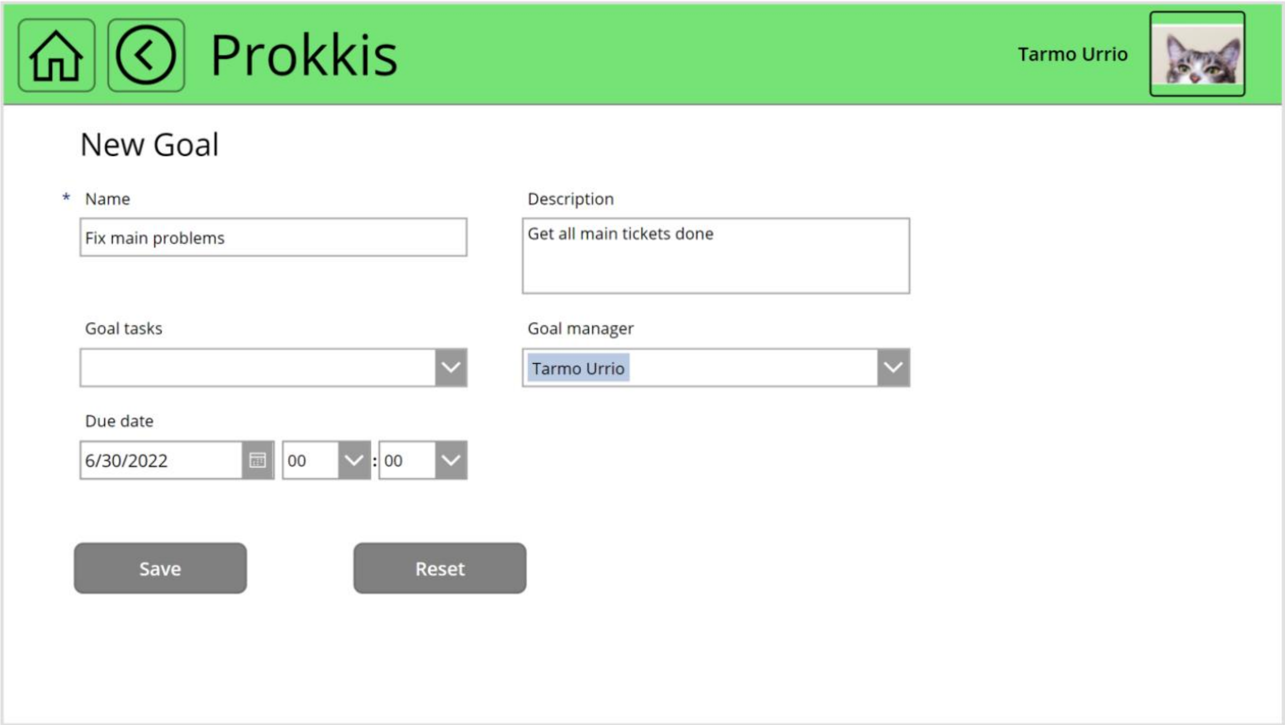


1/6/2022 - 1/8/2022
Client Project 101

Project for managing customer related tasks

ID: 1006

A new goal



* Name: Fix main problems

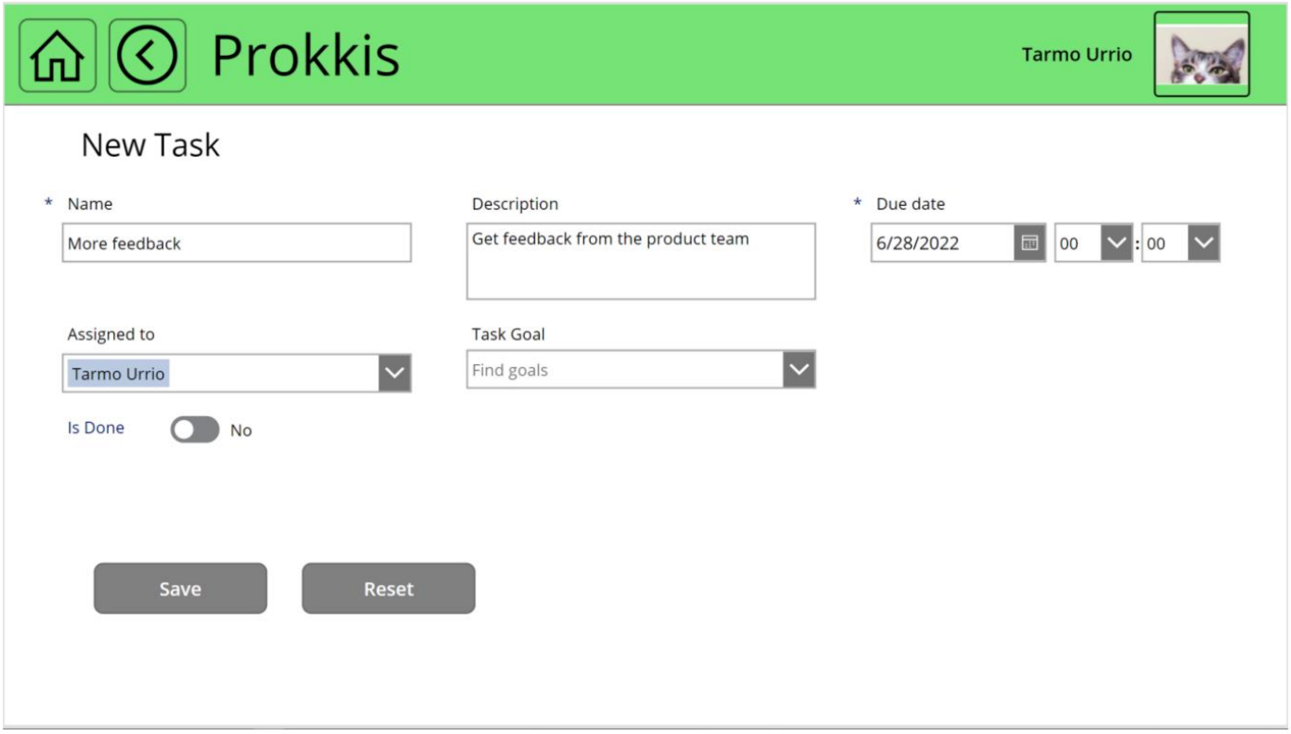
Description: Get all main tickets done

Goal tasks:

Goal manager: Tarmo Urrio

Due date: 6/30/2022 00 : 00

A new task



New Task

* Name: More feedback

Description: Get feedback from the product team

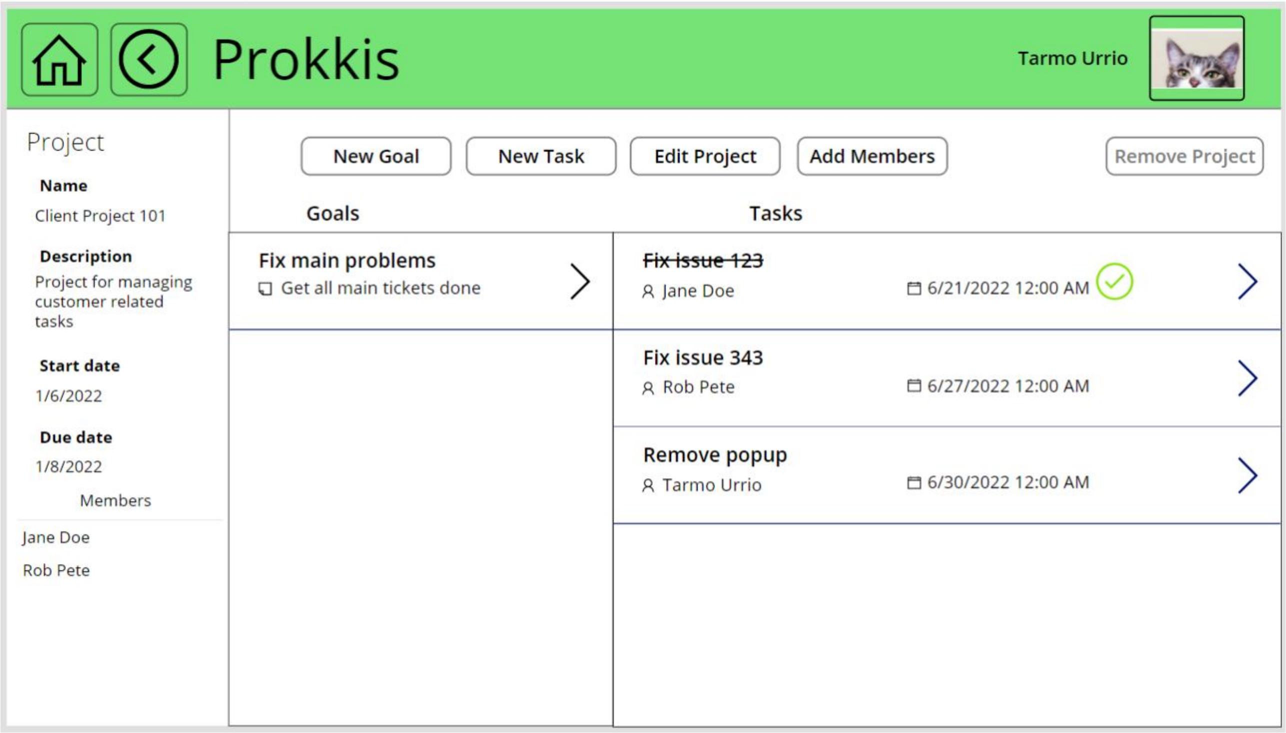
* Due date: 6/28/2022 00 : 00

Assigned to: Tarmo Urrio

Task Goal: Find goals

Is Done: No

Single project view



Project

Name
Client Project 101

Description
Project for managing customer related tasks

Start date
1/6/2022

Due date
1/8/2022

Members

Jane Doe
Rob Pete

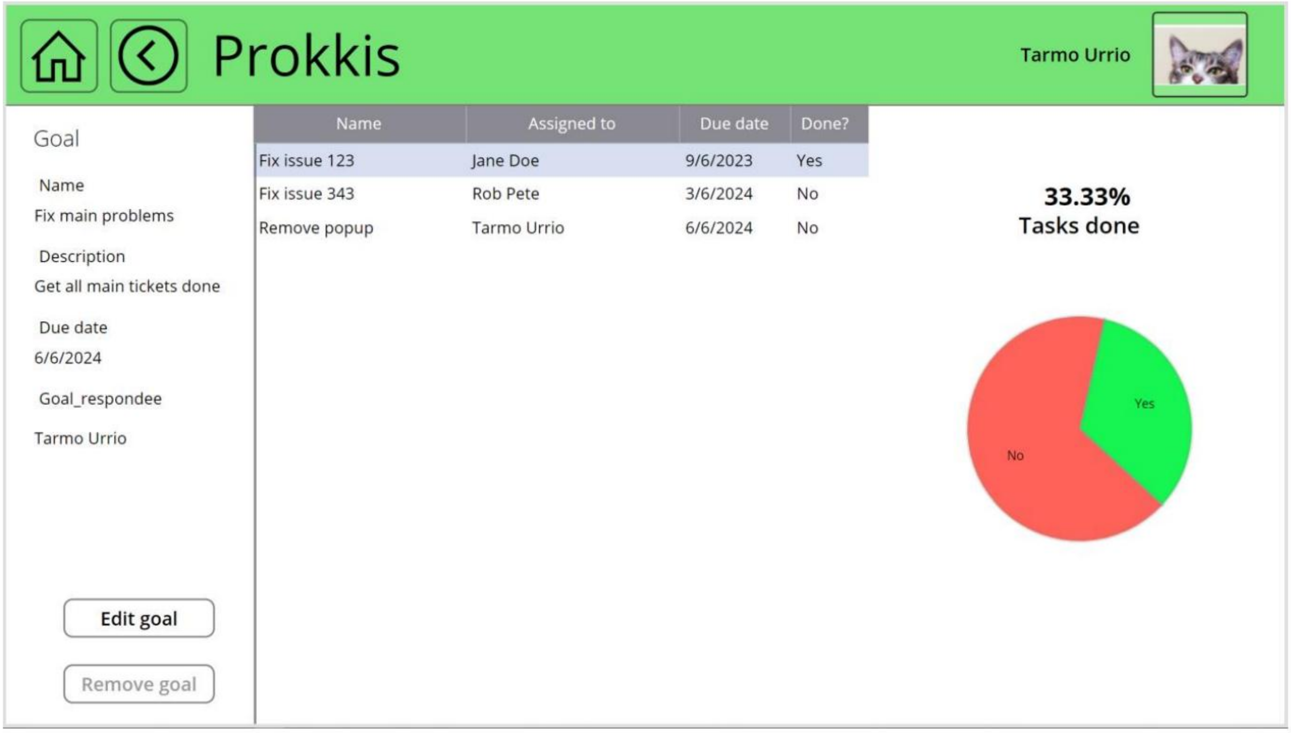
Goals

Fix main problems
Get all main tickets done

Tasks

~~Fix issue 123~~	Jane Doe	6/21/2022 12:00 AM ✓
Fix issue 343	Rob Pete	6/27/2022 12:00 AM
Remove popup	Tarmo Urrio	6/30/2022 12:00 AM

Single objective view



Goal

Name
Fix main problems

Description
Get all main tickets done

Due date
6/6/2024

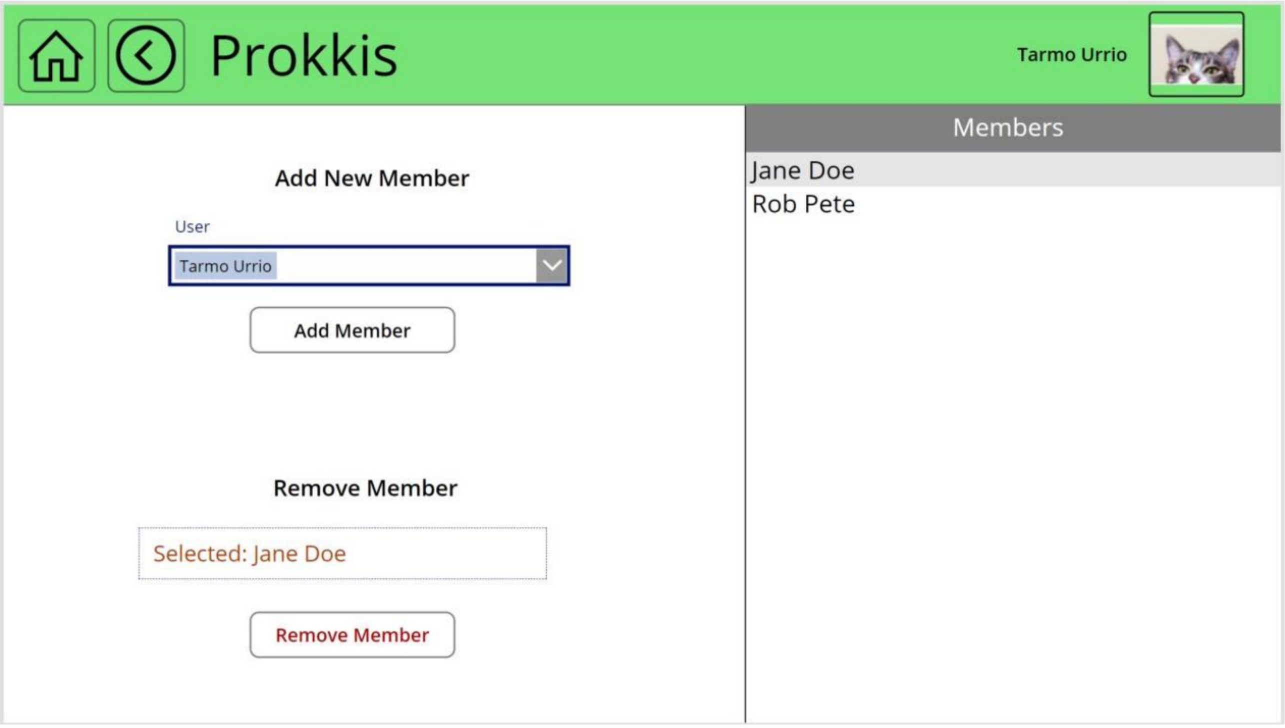
Goal_respondee

Tarmo Urrio

Name	Assigned to	Due date	Done?
Fix issue 123	Jane Doe	9/6/2023	Yes
Fix issue 343	Rob Pete	3/6/2024	No
Remove popup	Tarmo Urrio	6/6/2024	No

33.33%
Tasks done

Adding users to the project



Add New Member

User
Tarmo Urrio

Remove Member

Selected: Jane Doe

Members

Jane Doe
Rob Pete