https://doi.org/10.53656/math2024-5-2-fra

Science in Education
Научно-методически статии

FRAMEWORK FOR DESIGNING VISUALLY ORIENTATED TOOLS TO SUPPORT PROJECT MANAGEMENT

Dalibor Milev, Nadezhda Borisova, Elena Karashtranova
Faculty of Mathematics and Natural Sciences
South-West University "Neofit Rilski" – Blagoevgrad (Bulgaria)

Abstract. Modern technologies enable the use of visually oriented tools to organize projects, ideas, and tasks in a completely informal manner. Visually oriented software does not impose a structure or method for organizing and presenting user projects. This approach gives users the ability to organize their tasks in a visual format, promoting a more intuitive and adaptable work environment.

This article introduces software product designed and developed to effectively organize and plan tasks, projects, or ideas. The product allows users to create boards on which they can mark their ideas by adding certain elements. With these different elements, the workflow of each user can be visually illustrated and parts of the plan can be highlighted. Easy planning is achieved from an easy-to-read format with a variety of visual elements that are more natural to organize a plan with step implementation, and quickly share the plan for all of its development teams.

Keywords: visual board; visual software design; creative project management; collaborative platform

1. Introduction

Project planning and management are crucial aspects of software technologies. Many companies rely on software to streamline employee workflows. Such software typically offers functions, such as task planning and organization, team management, task assignment, and various other capabilities.

Different workflow planning models exist in software technology, such as Waterfall, Agile, Gantt, Spiral, and V-model (Khatri 2023). It is important to note that a model suitable for one project may not be suitable for another. Selecting the right model for a specific project is a crucial aspect of any well-organized project in software technology. For projects with a small number of employees, a model such as Big Bang (Half 2023) can be used, but this model is not recommended for projects with a larger scope.

Multiple software products with specific patterns are utilized to assist with work processes and task allocation. They are designed to support particular work processes and specific task allocation methods. However, some software products cater to more than one specific model. One commonly used model is Agile[1], which breaks down software product development into cycles, each involving goal-setting, development, planning, and testing. Agile is known for its flexibility and can be applied to a wide range of project types. Agile projects sometimes incorporate a method developed by Taiichi Ohno called "Kanban" (which is a visual map in Japanese)[2]. This method aims to represent tasks performed at a given time as a sequence of maps that illustrate the work process. The cards contain goals and tasks that are pending, currently being completed, or have already been completed. Maps present tasks in a visual form, organized into sections, in a way that is intuitive and easy to understand.

The use of visually oriented software elements allows easy creation of projects, organizing a plan with execution steps, and sharing the plan among participants in the development process. Users can define the arrangement of the elements within the project. Examples include workflow illustrations, block diagrams of algorithms, links between website modules, scene planning, calendar tasks, and marketing strategies.

One of the main goals is to provide a fast, easy, intuitive, and sustainable approach to planning or organizing through which the user can quickly change or add items.

One of the main advantages of such a platform is that the freedom of organization allows teams from a wide range of businesses and industries to use it.

This study presents a software product designed to effectively organize and plan tasks, projects, and ideas. The product provides an opportunity for users to create boards on which they can mark their ideas by adding certain elements. With these various elements, each user's workflow can be visually illustrated, and parts of their planning can be highlighted.

The paper is organized as follows: The introduction is followed by a discussion of the types of software based on their approach for organizing and structuring the work of their users in Section 2. Section 3 outlines the process of design and development of creative project management software with conceptual and architectural models. Section 4 presents a summary and discussion of the results obtained. Finally, Section 5 concludes the paper.

2. Background

Planning is essential to successful project management, as it provides structure and clarity to the implementation process. It enables teams to set goals, allocate resources, and manage their time efficiently, minimizing risks and uncertainties (Bailey 2023). By creating detailed plans, teams can track progress and make adjustments in real-time, leading to higher productivity and better bottom lines. Planning also facilitates communication between team members and the client or end users by creating a shared vision and understanding of project goals. Whether using complex platforms or freer tools, good planning is fundamental to achieving success in any project.

A large number of software tools are used every day in large and small companies for coordination and planning, some of which are more restrictive to the user. The software itself, which is used for a project, be it personal or group, should provide users with the opportunity to realize the project in the most convenient way.

2.1. Project management software

The choice of software for a project depends on the number of participants, objectives, approach, and cost of the software. The project management and planning tools are numerous and have many different elements and functionalities.

There are different types of software based on their approach to organizing and structuring the work of their users.

2.1.1. Restrictive and structured software

This category includes software with a specific project planning approach defined by the methodologies and functionalities supported by the product itself. Many of these are aimed at large companies with numerous employees, requiring a structured approach for a safe and easy way to carry out all the steps and tasks. Examples of such platforms are (Patel 2024):

- **Microsoft Project** (Keup 2023) is a platform that has a wide range of functionalities such as determining resources, planning through graphs, calculation of data, resources, and costs. The platform is one of the largest and most used in the world.
- **Jira Align**[3] is widely used in software projects, but the flexibility of the tool itself allows it to be used in other types of projects as well. This platform contains extremely important functionalities such as task tracking, strong personalization through integrations with other tools, and detailed reports.

These two platforms are among the most widely distributed, especially in businesses and companies with a large number of employees.

2.1.2. Collaborative platforms

This category can affect quite a few types of platforms, with the purpose of the platform itself usually being to encourage collaboration and planning. In addition to offering more freedom than the previous category, these platforms provide the ability to schedule and report tasks visually. One of the most important aspects of project planning with tools of this kind is collaboration itself. Commenting, adding information, real-time updating, and other features allow teams to quickly and easily give and receive feedback. Examples of platforms of this type are:

- **Trello**[4] is an organized and simplified approach to project planning consisting of cards and groups in which tasks are organized (strongly related to the Kanban methodology). It can be defined as more orientated towards visual project management (Spolski 2024).
- **Asana**[5] provides options like relationships and dependencies between tasks and supports elements like Gantt chart and project life cycle bar.

2.1.3. Creative project planning

Creative project planning is about visual expression in many different ways that mostly depend on the users themselves. Such tools cannot be categorized only as project management and planning tools, as the freedom of their design allows them to be used for many other purposes as well. One of the main characteristics of such platforms is that they are universal and can be used to implement a single task or to create a visual project timeline. Some users may fully utilize the project planning platform, although such tools are not suitable for use in teams with large numbers of employees. Two examples of platforms of this type are:

- **Milanote**[6] is a platform with an interactive and intuitive interface. It allows its users to easily create and share their projects. The illustration of design tasks does not necessarily follow the convention provided but can be chosen by the user.
- **Monday.com** provides a large number of templates and customization options, allows users to track progress, and efficient scheduling through a timeline (Lambden 2024).

Each of these three types of project organization and planning can be used to implement real projects involving employees, resources, and many other elements. Not every approach is suitable for a particular project, and different projects are easier to implement with one type than with another. The nature of the project or task determines whether the implementation and planning will fit the tool being used.

Any of the three approaches can produce positive results. Project evaluation, number of team members, methodology, and many other elements influence whether project execution using a particular planning platform will generate positive results.

A project that is implemented with a small team and includes many elements that are not yet defined or will require enhancement and extension of descriptions will be easier to implement using one of the tools designed for quick and easy data exchange, which gives more freedom. On the other hand, large teams require a more precise organization, which makes restrictive platforms more suitable for this type of project. Table 1 summarizes the main differences between restrictive and creative project planning methods (PPM).

Table 1. Differences between restrictive and creative PPM

	Restrictive and structured software	Creative project planning
Simplicity of modification	They work with specific software development models, providing various options and functionalities tailored to those methods.	They do not impose restrictions on approaches and give complete freedom without restricting users with methodologies or structures.
Management	Accuracy is achieved through detailed and organized project planning, resource inclusion, and task assignment to employees.	They are not suitable for large teams and complex projects. Collaborating on a project can become burdensome with many people involved.
Visually represent tasks	Options for visually presenting tasks in a timeline, or through graphs, to illustrate the relationship between completed tasks, those not yet started, and other options.	They lack functionalities found in highly structured platforms, such as detailed project timeline planning and tracking.
Interface	The structure of these tools can hinder creative thinking about projects from various angles, making tasks and processes more complex.	Easy to create user profiles, easy to create and link to a project. Real-time collaboration where any team member can add comments, remarks, add items to the conversation itself, etc.
Price	Higher price, which for smaller companies can turn out to be a big risk.	Prices for products of this type are lower or free in certain cases, which is a preferred option for startups, smaller projects, or small teams.
Learning	Provide an overly complex and busy interface that makes it difficult for new users, and the learning process can slow down the workflow.	They have an intuitive and easy-to-understand design that allows people without technical training to participate in the implementation of projects with this tool.

Both of types software presented in Table 1 provide functionalities and services that can be used to implement a project in a real environment. Highly specialized programs, the purpose and structure of which are created for project management, are very common and find their application in most firms and companies. However, creative project planning software is also used by large companies for describing processes and considering solutions, as well as in smaller projects.

3. Developing creative project management software

The following aspects are important for the implementation of a well-organized platform:

• It is important for users in different professions to note that organization, simplicity of the platform, and visual representation of tasks are crucial.

• Visual elements help with easy and quick orientation in project planning and organization.

• The platform should not be limited to a specific user group and should be available for a variety of projects.

• It is essential for team members to have access to the visual interface at all times and be able to contribute to the workflow in dynamic fields and projects involving personal tasks and goals.

In conclusion, the development of new platforms, software, or tools for organizing and managing creative projects should have an easy and intuitive design that allows for a basic visual presentation of tasks, ideas, concepts, and projects. When creating applications in an industry with highly structured platforms, selecting an appropriate and realistic target group is critical.

The selection of technologies for a project management software is a crucial decision, as it determines the approach to achieving the project's objectives. Key objective for the user interface of such tool is to implement dynamic features therein, i.e. to include moving elements, resizing elements, facilitating the interaction between two elements, drawing on a canvas, and rotation. It could be achieved by using the Interact.js library. This library provides key functions for creating dynamic applications or website interfaces for organizing and managing tasks. Some of the functions offered by the library include moving elements, resizing elements, facilitating interaction between two elements, drawing on a canvas, and rotation. Interact.js is an open-source library that allows users to selectively use only the necessary functions, thereby reducing file size. It also supports event handling and is compatible across different web browsers.

For our development process, we utilize several integrated development environments: Blazor Server App, Microsoft SQL Server Management Studio for database management, and Microsoft Visual Studio for page management.

Blazor is a web-based framework that uses .Net and C# and provides a deployment option, as it supports server and client-side build and configuration[7]. The two parts are built sequentially in Blazor, allowing for easier integration between them, and programming is entirely based on a single programming language. The framework uses Razor components, whose structure includes HTML and C# code, but also allows JavaScript libraries to be included in projects.

The .NET framework supports an option that does not allow an intermediary member to exist between the server and the client when the connection is not HTTPS[8]. This approach is not mandatory but provides a level of organization that is necessary not only for projects of this kind, but for any kind that contains personal user information.

The Microsoft Visual Studio platform is also used for project development. This development environment provides a wide range of templates whose structure can be used for different types of projects[9], but for this project, the template used is the Blazor Server App. Apart from these functionalities, it also allows connection with Git, Github, Azure, and many other platforms. Integration with them allows collaboration, code management and branching, etc.

Using .NET and Blazor in the Microsoft Visual Studio development environment is a popular choice and one of the most common methods for developing these types of projects.

MS SQL Server Management Studio provides functionalities that are required in database management, as users are required to be able to save their projects, tasks, etc. The additional capabilities of visualizing the schemes, tables, and dependencies between them, adding roles, authorization, and other functionalities, are essential to maintain the security of the project. Integration with Visual Studio[10] is also one of the important reasons we chose this tool. For the implementation of the project, an approach[11,12] was used to design the database through the Entity framework, which allows not to use SQL queries but to interact with the database, tables, and records through C# objects or LinQ queries.

3.1. Conceptual model

The user interface of the PinboardPM website communicates with the database using Entity framework. The creation, editing, and deletion of data and records in the tables are performed at runtime of the application using defined functions. The interactions of the above mentioned technologies are illustrated in fig. 1.

3.1.1. Database

The database includes the following tables *Users*, *Pages*, *htmlElements* and *Images*. Relationships are defined between the tables to ensure proper data storage and project organization.

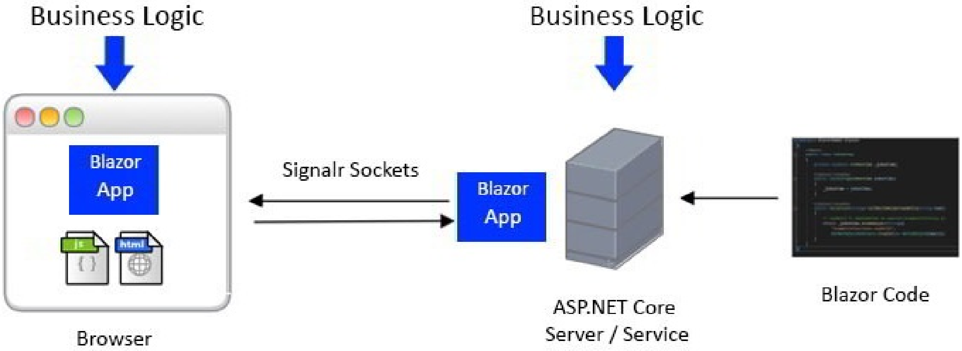


Figure 1. Mobile App Development Using Blazor (taken from (Doshi 2020))

3.1.2. Client Side

The front-end of the application is built using Razor pages. Their structure shares certain elements, such as a navigation menu or the main view of pages, and these elements and structures are saved in separate files. Each Razor page has an HTML structure and a C# code section that contains functions, variables, and other objects needed to perform the functionality.

3.1.3. Server Side

Functionalities to create profiles, log in users, or edit and delete database data are performed with defined classes or through the capabilities of the .NET Framework[13], the Entity Framework, and the classes *UserService*, *AuthenticationService* and *DbContextClass*[14].

3.2. Architectural model

The connections between the main modules of the PinboardPM system are presented in fig. 2. Our system is designed for creative teams and should offer an intuitive and user-friendly interface. The proposed architectural model is based on two user roles that can be used in the PinboardPM system: “administrator” and “user”.

Administrators have access to the administration after logging in to the system, as well as to the pages of their projects. The user who only has the “*user*” role does not have access to the administration and only to the pages created by him/her.

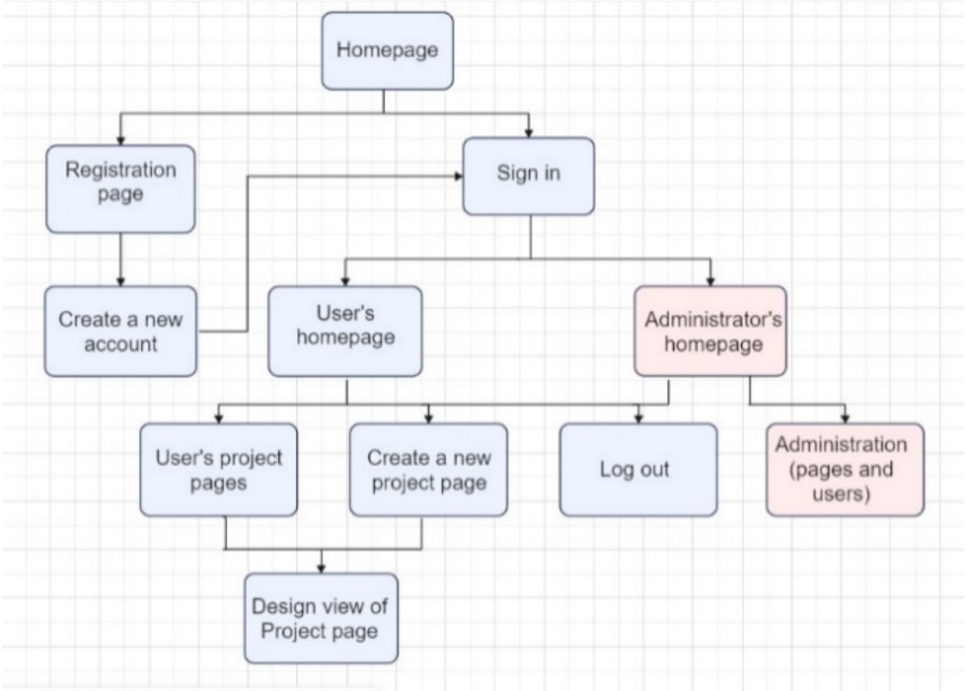


Figure 2. Architectural model of PinboardPM

The Blazor Server App template is used to implement a larger part of the project on the server without loading the client part. This kind of design allows the application to communicate between the server and the client in real time, without delay, and the loading of the application itself is fast. The template allows the client side to be off-loaded with the application logic, to protect the data and information.

The project allows users to create new pages for different projects or tasks. To achieve this, the pages table in the database (fig. 3) stores not only the primary keys that represent each page but also a foreign key of the user. This key is defined by the one-to-many relationship, meaning that one user can create many pages, but one page can only be owned by one user.

Each page that is created shows a virtual board where different elements to represent the user's tasks can be ordered. The htmlElements table is used to store the data of these elements. Entries in the table are updated dynamically during program execution using various functions. Each element has a unique identifier as well as a foreign key that stores the identifier of the page the element belongs to. The goal is that when one of the pages is loaded, its interface returns a list of all elements that have a page key stored, and then loads them. The attributes Left and Top represent the X and Y coordinates of the element in numerical form, while Width and Height refer to the size of the element in pixels.

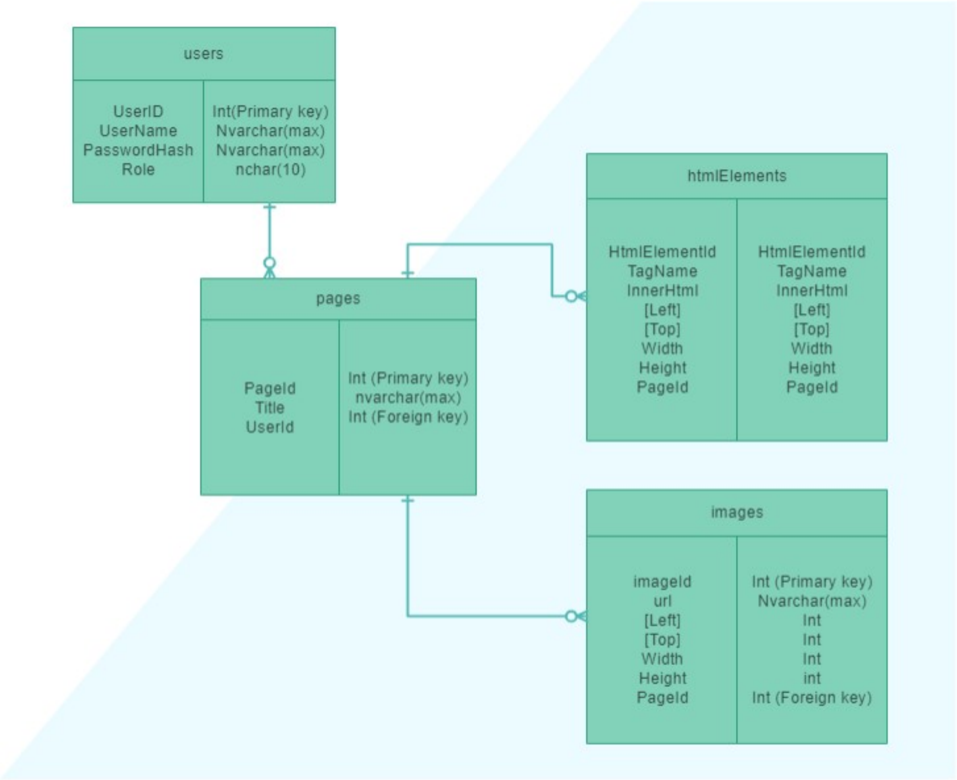


Figure 3. Relational database of PinboardPM

The table structure htmlElements is not suitable for all types of elements. To efficiently store image data, we use an additional table called images. The images table is specifically used to store the image data that users can add to the application interface.

The users table stores data about the user's ID, the user's name (no repetition of two identical names is allowed), the user's password (entered during registration), and a role that determines whether the user has administrative access.

The next step in project implementation involves creating models that represent the tables. These models will be used by the class DbContextClass to create the database and its tables. When defining the class that represents a table, member variables are declared for each table field. Listing 1 presents one of the four models used to create the tables. The structure of the other three is identical – the table fields are declared as member variables of the class with a name representing the table.

Listing 1. Model ClassUser

```
public class ClassUser
{  [Key]
   public int UserID { get; set; }
   public string Username { get; set; }
   public string PasswordHash { get; set; }
   public string Role { get; set; }
}
```

In the *DbContextClass* class (Listing 2), lists of objects for the four models are defined, as well as LinQ queries that create the relationships between the tables.

Listing 2. Model DbContextClass

```
public class DbContextClass : DbContext
{ public DbContextClass(DbContextOptions<DbContextClass>
         options) : base(options)
   public DbSet<ClassUser> Users { get; set; }
   public DbSet<Page> Pages { get; set; }
   public DbSet<HtmlElement> htmlElements { get; set; }
   public DbSet<Image> images { get; set; }
   protected override void
   OnModelCreating(ModelBuilder modelBuilder)
   {  modelBuilder.Entity<HtmlElement>()
        .HasOne<Page>()
        .WithMany(p => p.HtmlElements)
        .HasForeignKey(e => e.PageId);
      modelBuilder.Entity<Image>()
        .HasOne<Page>()
        .WithMany(p => p.Images)
        .HasForeignKey(i => i.PageId);
    }
}
```

Once the models are created, we implement functionality that enables dynamic moving and resizing of elements. To achieve this, we utilize the free open-source library, Interact.JS[15]. The function is defined in a separate JavaScript file (refer to Appendix A) and includes all the necessary functionalities for the project:

• Moving elements: When a user moves an element on the board, the application calculates the new position of the element relative to the position of the cursor. The functionality also includes the restriction that the move is only possible within the boundaries of the elements that have the *main-content* class, keeping items within the virtual dashboard.

• Resizing of elements: This functionality allows for dynamic resizing of the added elements. Appendix A lists the options for changing the width or height of an element as they are relevant to the customization of user projects.

• Adding element functionality: For this purpose, the *draggable* class has been added to elements that the user can move.

4. Results and analysis

The home page of the application is available to all users. Full functionalities for creating a new page or reviewing a created page are available only to registered users (fig. 4).

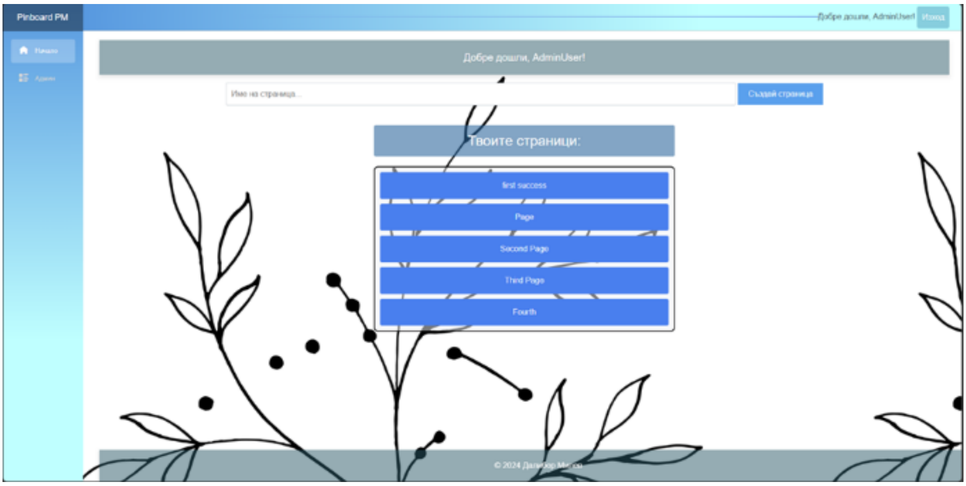
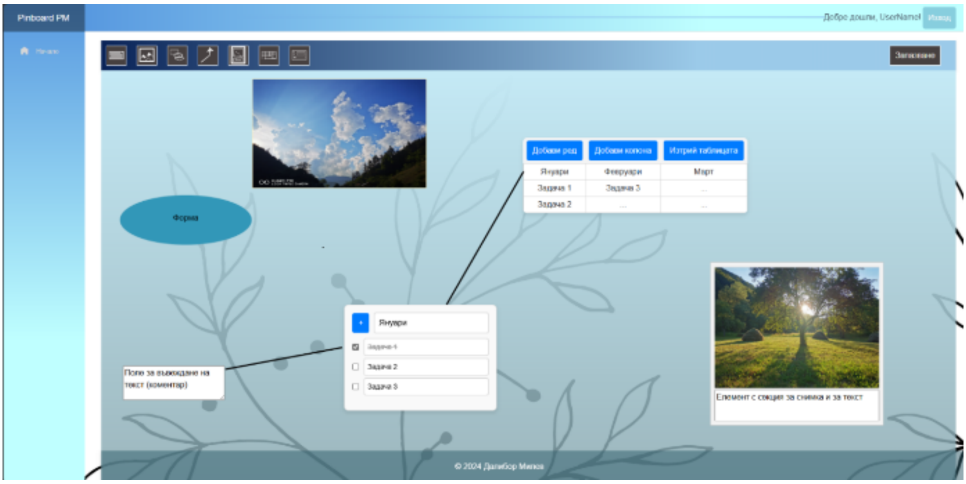


Figure 4. Home page for registered users

The home page for registered users is available after successful login. It allows the user to create a new project page or to view or modify a page he has created by selecting from the list of pages whose records in the database contain his ID number. Creating a new page or modifying an already created page opens a Pinboard page. The Pinboard page (fig. 5) contains the virtual board on which users can plan their ideas, projects and tasks. Buttons execute functions whose role is to add various types of html elements to the space marked with the `class = ’’main-content’’` attribute.



Януари	Февруари	Март
Задача 1	Задача 3	...
Задача 2	...	...

Форма

Януари

☑ Задача 1

□ Задача 2

□ Задача 3

Поле за въвеждане на текст (коментар)

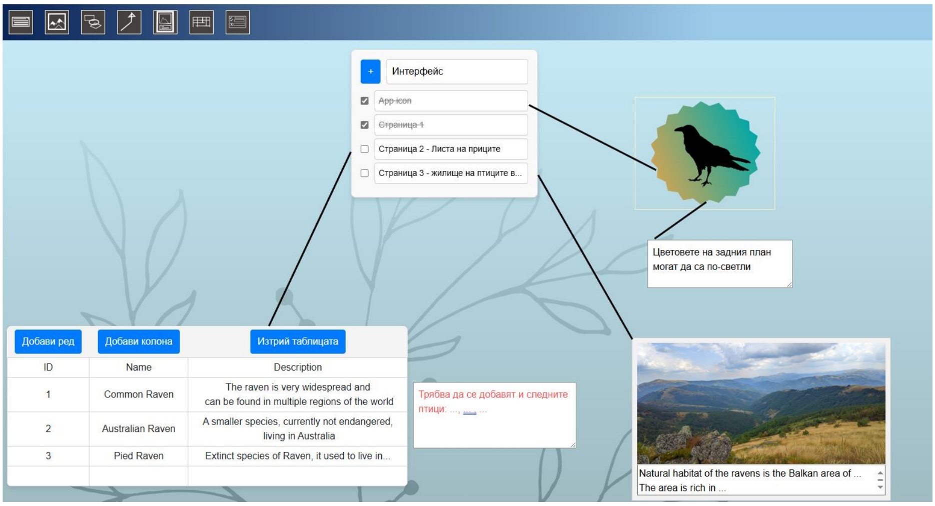
Елемент с секция за снимка и за текст

© 2024 Далибор Милев

Figure 5. An example of visual elements on the virtual board

The buttons are equipped with tooltips that appear when the cursor hovers over them. When a button is clicked, it triggers one of the following functions:

- `addTextBox()` – adds an element of type textarea.
- `addImage()` – adds an image.
- `addStyledDiv()` – creates a div element in which the user can enter text and change its shape between square (rectangle) and circle.
- `addDynamicDiv()` – creates a div element with two sections – one for text and the other for an image.
- `addTable()` – adds a table, with the ability to add or remove rows and columns dynamically.
- `addRadioList()` – creates a section with a title field and a button to add a row (task) to the list. Each task includes a checkbox that marks it as complete.
- `startDrawing()` – responds to various events (mousedown, mousemove, mouseup) and executes specific code based on these events. It allows drawing on a virtual board by creating a div element representing a line[16] between the mousedown and mouseup events.



Интерфейс

☑ ~~App icon~~

☑ ~~Страница 1~~

□ Страница 2 - Листа на птиците

□ Страница 3 - жилище на птиците в...

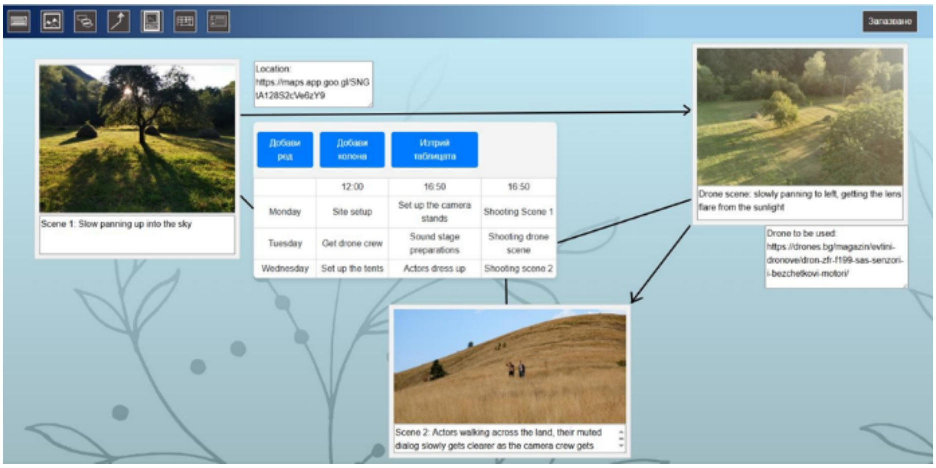
Цветовете на задния план могат да са по-светли

ID	Name	Description
1	Common Raven	The raven is very widespread and can be found in multiple regions of the world
2	Australian Raven	A smaller species, currently not endangered, living in Australia
3	Pied Raven	Extinct species of Raven, it used to live in...

Трябва да се добавят и следните птици: ..., ..., ...

Natural habitat of the ravens is the Balkan area of ...
The area is rich in ...

Figure 6. Mobile Application Development Project



Scene 1: Slow panning up into the sky

Location:
https://maps.app.goo.gl/SNGtA128S2cVe6zY9

	12:00	16:50	16:50
Monday	Site setup	Set up the camera stands	Shooting Scene 1
Tuesday	Get drone crew	Sound stage preparations	Shooting drone scene
Wednesday	Set up the tents	Actors dress up	Shooting scene 2

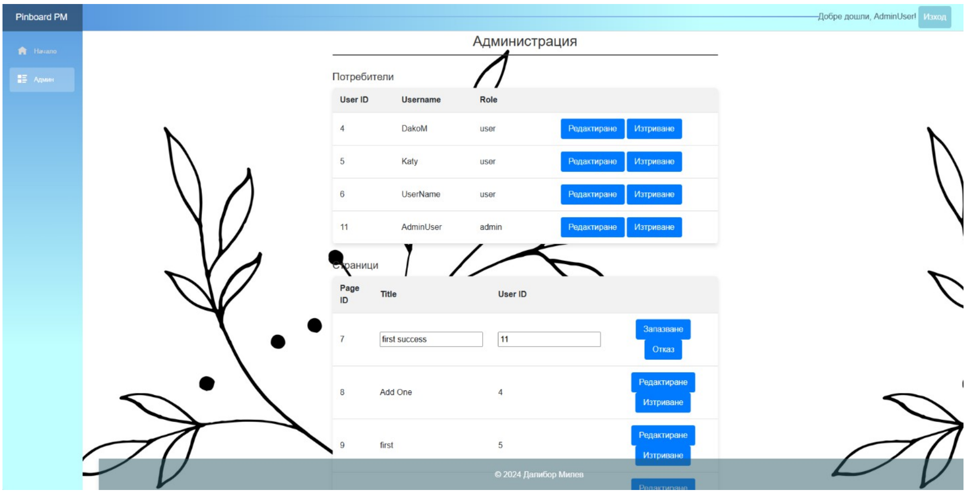
Drone scene: slowly panning to left, getting the lens flare from the sunlight

Drone to be used:
https://drones.bg/magazin/evtini-dronove/dron-zfr-f199-sas-senzori-i-bezchetkovi-motori/

Scene 2: Actors walking across the land, their muted dialog slowly gets clearer as the camera crew gets

Figure 7. Planning to shoot scenes

- UpdatePage() – preserves the modified positions and sizes of the elements and updates this data in the database tables. It makes use of the htmlElements and images lists from the DbContext class to locate the elements with the page ID. These elements are then updated with new values in their fields and the changes are saved.



Администрация

Потребители

User ID	Username	Role	
4	DakoM	user	Редактиране Изтриване
5	Katy	user	Редактиране Изтриване
6	UserName	user	Редактиране Изтриване
11	AdminUser	admin	Редактиране Изтриване

Страници

Page ID	Title	User ID	
7	first success	11	Запазване Отказ
8	Add One	4	Редактиране Изтриване
9	first	5	Редактиране Изтриване

Figure 8. Administration page

The implemented functionalities allow the PinboardPM website to be used to organize and manage real tasks and projects. Website users can create their pages on which to add and organize visual elements to illustrate their tasks. Fig. 6 shows the planning of the mobile application development project, while fig. 7 shows the planning to shoot scenes. Both scenarios are related to the creative planning stage of the user. An administration page (fig. 8) has been created to make it easier to maintain the website and manage user data and their pages. This page allows editing and deleting records from the `Users` and `Pages` tables in the database. The tables on the Administration page are dynamic and show the number of records for each of the Users and Pages tables in the database.

5. Conclusion

The first step in implementing any project is planning and organizing related activities and tasks. The developed Pinboard website offers project-planning capabilities for creative users or small teams in an unstructured and non-restrictive way. The application uses the author's photos as examples and icons for the buttons. The user interface utilizes dynamic element scrolling, enabling users to create their own pages. The database is updated at runtime, and services are used to perform checks for various functions such as “Login”, “Registration”, and “Administration”.

To validate the design and development approach of the Pinboard application, a demo version of a project planning system was created. The software caters to two types of users: administrators and regular users. It allows for the creation and updating of project activities, user data management, and data security.

The website provides elements that enable users to organize the implementation of their projects and ideas using the application’s flexible functionalities. These features work together, allowing users to manage their tasks easily and quickly, and present their projects, goals, and tasks efficiently and creatively.

Teamwork offers different perspectives and is a crucial part of executing activities within any project. Therefore, the user team should be involved in this process, with each member having access to the planning and organization of various activities. Currently, users can only utilize the software for their projects through their profiles. However, we are planning to introduce the option for team accounts in the next version of the system. At the time of writing this article, the proposed software is under review by experts in the field of graphic design. We have also planned to conduct questionnaires to assess the satisfaction of specialists with the software product.

APPENDIX

Listing 3. Function for moving and resizing elements

```
function dragAndDrop(className) {
 interact(className)
  .draggable({
   modifiers: [
    interact.modifiers.restrictRect(\{restriction:
        ’.main-content’},),],
    listeners: {
     move(event) {
      const target = event.target;
      const x = (parseFloat(target.getAttribute(’data-x’))
                 || 0) + event.dx;
      const y = (parseFloat(target.getAttribute(’data-y’))
                 || 0) + event.dy;
      target.style.transform = ‘translate(${x}px, ${y}px)‘;
      target.setAttribute(’data-x’, x);
      target.setAttribute(’data-y’, y);
                 },
               },
   }).resizable({
    edges: { left: true, right: true, bottom: true, top: true },
 modifiers: [
   interact.modifiers.restrictEdges({
     outer: ’.main-content’,
}),
interact.modifiers.restrictSize
                        ({min:{width:50,height:50},}),],}).
on(’resizemove’, (event) => {
  const target = event.target;
  const dx = event.deltaRect.left;
  const dy = event.deltaRect.top;
  const newWidth = event.rect.width;
  const newHeight = event.rect.height;
  target.style.width = ‘${newWidth}px‘;
  target.style.height = ‘${newHeight}px‘;
  target.style.left=‘${parseFloat(target.style.left)+dx}px‘;
  target.style.top = ‘${parseFloat(target.style.top)+dy}px‘;
  });
}
```

NOTES

1. The Agile Coach. *What is the Agile methodology?.* https://www.atlassian.com/agile (accessed Dec. 23, 2023).
2. Kanban Tool. *History of Kanban.* https://kanbantool.com/kanban-guide/kanban-history (accessed Jan. 15, 2024)
3. Atlassian. *Achieving Together What’s Impossible Alone.* https://www.atlassian.com/customers (accessed Feb. 23, 2024)
4. Atlassian. *5 Trello features that will change the way you work.* https://www.atlassian.com/blog/trello/trello-features-change-the-way-you-work (accessed Dec. 10, 2023)
5. Asana Help Center. *All Asana features.* https://help.asana.com/hc/en-us/articles/14109618961435-All-Asana-features (accessed Dec. 20, 2023)
6. Milanote Review. *Milanote Details, Pricing and Features.* https://www.cloudfindr.co/software/milanote/ (accessed May 10, 2024)
7. Microsoft Learn. *ASP.NET Core Blazor.* https://learn.microsoft.com/en-us/aspnet/core/blazor/?view=aspnetcore-8.0 (accessed March 08, 2024)
8. Microsoft Learn. *Enforce HTTPS in ASP.NET Core.* https://learn.microsoft.com/en-us/aspnet/core/blazor/?view=

aspnetcore-8.0 (accessed March 08, 2024)

9. Microsoft. *Visual Studio: IDE and Code Editor for Software Developers and Teams.* https://visualstudio.microsoft.com/ (accessed April 20, 2024
10. Microsoft Learn. *SQL Server Management Studio (SSMS).* https://learn.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver16 (accessed April 20, 2024)
11. Microsoft Learn. *Overview of Entity Framework Core – EF Core.* https://learn.microsoft.com/en-us/ef/core/ (accessed May 05, 2024)
12. Microsoft Learn. *Querying Data – EF Core.* https:// learn.microsoft.com https://learn.microsoft.com/en-us/ef/core/querying/ (accessed May 12, 2024)
13. Wikipedia. *.NET Framework.* https://en.wikipedia.org/wiki/.NET_Framework (accessed April 08, 2024)
14. Microsoft Learn. *DbContext Class (System.Data.Entity).* https://learn.microsoft.com/en-us/dotnet/api/system.data.entity.dbcontext?view=entity-framework-6.2.0 (accessed May 12, 2024)
15. interact.js. *JavaScript drag and drop, resizing and multi-touch gestures for modern browsers.* https://interactjs.io/ (accessed April 5, 2024)
16. KennyKee Styles. *Drawing A Line Between Two Draggable DIVs.* https://kennykee.com/128/drawing-a-line-between-two-draggable-divs/ (accessed May 15, 2024)

REFERENCES

BAILEY, G., 2023. *Famous Project Management Failures and What to Learn from Them.* https://www.prosymmetry.com/resources/4-famous-project- management-failures-and-what-to-learn-from-them/

DOSHI, M., 2020. *Microsoft Blazor Enables Native App Development in C# and .NET.* https://www.semaphore-software.com/microsoft-blazor-enables-native-app- development-in-c-and-net/

HALF, R., 2023. *6 effective SDLC models: Which one is best?* https://www.roberthalf.com/au/en/insights/management-tips/6-basic-sdlc-methodologies-which-one-best

KEUP, M., 2023. *What Is Microsoft Project? Uses, Features and Pricing.* https://www.projectmanager.com/blog/what-is-microsoft-project

KHATRI, V.S., 2023. *Top 7 SDLC Methodologies: Phases, Models and Advantages.* https://hackr.io/blog/sdlc-methodologies

LAMBDEN, D., 2024. *monday.com Review: Pros, Cons & 2024 Platform Test Results.* https://tech.co/project-management-software/monday-review (accessed May 10, 2024)

PATEL, R., 2024. *70+ Software Development Statistics and Facts.* https://radixweb.com/blog/software-development-statistics

SPOLSKY, J., 2024. *Announcing Trello.* https://www.joelonsoftware.com/2011/09/13/announcing-trello/

✉ **Dalibor Milev, Student**
Faculty of Mathematics and Natural Sciences
South-West University "Neofit Rilski"
Blagoevgrad, Bulgaria
E-mail: `dalibor.milev14@gmail.com`

✉ **Dr. Nadezhda Borisova, Assist. Prof.**
ORCID iD: 0000-0001-6956-5681
Faculty of Mathematics and Natural Sciences
South-West University "Neofit Rilski"
Blagoevgrad, Bulgaria
E-mail: `nborisova@swu.bg`

✉ **Dr. Elena Karashtranova, Assoc. Prof.**
ORCID iD: 0000-0003-0349-6171
Faculty of Mathematics and Natural Sciences
South-West University "Neofit Rilski"
Blagoevgrad, Bulgaria
E-mail: `helen_k@abv.bg`