



COMPE 494 Senior Project Report

Personal Task Organizer on the Web (PerTaskWeb)

Team Members: Salih Eren TAŞTAN, Furkan YILDIZ, Hasan Berk KELEŞOĞLU

Course Coordinator : Prof. Dr. Ali YAZICI

Supervisor : Dr. Bilge SAY

January 15, 2019



Abstract

Personal Task Manager on the Web (PERTASKWEB) is a practical task organizing tool for all type of users. It can be used as a task manager, for common users or business oriented users, as it has the capability to work as a plain manager, in which user's tasks can be saved and organized in hierarchical form. This tool will also help the user to organize all his/her tasks (work related or not) in a tree hierarchy and schedule them if needed. The user will be able to move the tasks from scheduled tree list to the unscheduled tree list, and vice versa.

Beside the hierarchical organization of the tasks, user will organize his tasks in different categories, which he/she previously creates. Also tasks can be sorted according to their attributes, like creation time, priority, due date, etc.

PERTASKWEB is a software tool which will be designed to help users manage their daily tasks using a web browser. PERTASKWEB is a web based system running on a web server. This system has been developed in PHP, JavaScript, bootstrap, jQuery and Ajax technologies are also utilized. The system do not need any installation of a database management system. The system can be installed to any PHP supported web server. The main idea of the system is to provide users a personal task organizer software tool on the web.

Keywords

Task organizing, task categories, task scheduling, web database, PHP, JavaScript, SQLite



Acknowledgements

We would like to express our deepest appreciation to all those who provided us the possibility to complete this project. We thank **Dr. Akın Kağan AKÇALI** for the interview on the mind mapping and task organizing as well as the relationship between them. A special gratitude goes to our project supervisor, **Dr. Bilge SAY**.



Table of Content

Abstract	2
Keywords	2
Acknowledgements.....	3
1. Introduction	8
2. Literature review	9
<i>2.1 Existing Desktop Software Tools for Task Management.....</i>	<i>10</i>
2.1.1 Task Coach	11
2.1.2 Task Unifier	11
2.1.3 Swift To-Do-List.....	12
2.1.4 Leader Task.....	12
2.1.5 My Life Organized	13
2.1.6 Ever Note.....	13
2.1.7 Comparison of Desktop Software Tools for Task Management	14
2.1.8 Comparison of Web-based Software Tools for Task Management.....	16
3. Requirements	19
<i>3.1 Overall Description of the Project</i>	<i>19</i>
3.1.1 Product Perspective.....	19
3.1.2 Product Functions.....	19
3.1.3 User Characteristics	20
3.1.4 Constraints	20
3.1.5 Assumptions and Dependencies	20
<i>3.2 Specific Requirements</i>	<i>20</i>
3.2.1 Main	20
<i>3.2.1.1 Minimize to tray.....</i>	<i>20</i>



3.2.1.2 <i>Exit</i>	21
3.2.2 Category	21
3.2.3 Task	22
3.2.4 Calendar	26
3.2.5 View	28
3.2.6 Register, Login and Log Out	29
3.2.7 Implementation Results of the Functional requirements	30
3.3 <i>Non-functional Requirements</i>	32
3.3.1 Database Requirements	32
3.3.2 Hardware Interface Requirements	34
3.3.3 Software Interface Requirements	34
3.3.4 Communications Interface Requirements	34
3.3.5 Performance Requirements	34
4. Design.....	35
4.1 <i>Software Methodology</i>	35
4.2 <i>System Architecture</i>	35
4.3 <i>UML Diagrams of the System</i>	36
4.3.1 Activity Diagram	36
4.3.2 Components Diagram	37
4.3.3 Deployment Diagram	37
4.3.4 Communication Diagram	38
4.3.5 Use Case Diagram	39
4.4 <i>User Interface Design of the System</i>	40
4.4.1 <i>Login and Register Pages</i>	40
4.4.2 <i>Main Menu- Home Page</i>	41
4.4.3 <i>Main Menu- Summary</i>	41
4.4.4 <i>Main Menu- ToDo List Page</i>	42



4.4.5 Main Menu- ToDo List Page (<i>ToDo List for the selected User</i>)	42
4.4.6 Main Menu- ToDo List Page (<i>ToDo List for the selected User Group</i>).....	43
4.4.7 Main Menu- Add Project Page.....	43
4.4.8 Main Menu- Calendar Page (<i>for the tasks of a selected user in selected month</i>).....	44
4.4.9 Main Menu- Calendar Page (<i>for the tasks of a selected user group in selected month</i>).....	44
4.4.10 Main Menu- Summary Page (<i>Select a Project</i>).....	45
4.4.11 Project / Tasks Menu- Edit Project Page (<i>Submit Changes</i>)	46
4.4.12 Project / Tasks Menu- Add Task Page	47
4.4.13 Main Menu- Summary Page (<i>Select a Task or Subtask</i>)	48
4.4.14 Project / Tasks Menu- Edit Task Page (<i>Submit Changes</i>)	49
4.4.15 Task Groups Menu- Add Task Group Page (<i>Add Task Group</i>)	50
4.4.16 Task Groups Menu- Manage Task Group (<i>Edit</i>).....	50
4.4.17 Task Groups Menu- Manage -Edit Task Group Page (<i>Delete</i>).....	51
4.4.18 Task Groups Menu- Manage Task Group Page (<i>After deleting a Task Group</i>).....	51
4.4.19 Users Menu- Add User Page	52
4.4.20 Users Menu- Manage User Page (<i>After adding a User</i>).....	52
4.4.21 Users Menu- Edit User Page.....	53
4.4.22 Users Menu- Manage Users Page (<i>After deleting a User</i>)	53
4.4.23 Users Menu- Show my Details Page.....	54
4.4.24 Users Menu- Edit my Details Page (<i>User Edit Page</i>).....	54
4.4.25 Users Group Menu- Add User Group Page.....	55
4.4.26 Users Group Menu- Manage User Groups Page	55
4.4.27 Users Group Menu- Edit User Group Page	56
4.4.28 Users Group Menu- Manage User Groups Page (<i>After deleting a User Group</i>).....	56
5. Implementation and Testing	57
5.1 Implementation.....	57



5.1.1 How to register.....	63
5.1.2 How to organize task categories.....	68
5.1.3 How to organize tasks	82
5.2 <i>Testing</i>	86
5.2.1 Test Plan	86
5.2.2 Test Procedures.....	86
6. Evaluation.....	87
6.1 <i>Problems</i>	87
6.2 <i>Performance</i>	87
6.3 <i>Usability</i>	87
6.4 <i>Deployment</i>	87
6.5 <i>Back up & Recovery</i>	87
6.6 <i>Security</i>	87
6.7 <i>Compliance with the Realistic Constraints</i>	88
7. Conclusion	89
References.....	90
Appendices:.....	91
<i>Appendix-A: Interview with Dr. Akin Kağan AKÇALI at HAVELSAN</i>	91
<i>Appendix-B: Sample Codes of the Sytems</i>	92
B.1 Login Codes	92
B.2 Main Codes.....	95
B.3. Category Codes	104
B.4 User Profile Codes.....	109



1. Introduction

Personal Task Manager on the Web (PERTASKWEB) is a software tool which will be designed to help users manage their daily tasks using a web browser. There are various desktop and web based products on the market that help people manage their calendars and tasks (e.g. Leader Task, My Life Organized, Evernote). However these tools are either calendar oriented or help you break down your projects and tasks in a mind mapping style.

Mind mapping is a highly effective way of getting information in and out of your brain. Mind mapping is a creative and logical means of note-taking and note-making that literally "maps out" your ideas. All Mind Maps have some things in common. They have a natural organizational structure that radiates from the center and use lines, symbols, words, color and images according to simple, brain-friendly concepts. Mind mapping converts a long list of monotonous information into a colorful, memorable and highly organized diagram that works in line with your brain's natural way of doing things. One simple way to understand a Mind Map is by comparing it to a map of a city. The city center represents the main idea (root node); the main roads leading from the center represent the key thoughts in your thinking process (child nodes); the secondary roads or branches represent your secondary thoughts, and so on. (*Akçalı, 2018*).

The interrelationship of calendar based view and the task based view of your projects and things to-do is not often emphasized. This tool will help the user to organize all his/her tasks (work related or not) in a tree hierarchy and to schedule them if needed.



2. Literature review

Organizing daily, weekly... not only routines, plans etc. but also our dreams or wishes that we want to do in our lives is common need for nearly all of people. Because no body wants to miss anything in their lives. So, people take notes or hire an employee for remind them to their plans. However, even if they have personal assistant or daybook, they might miss something that was scheduled before. But as everyone knows that nobody leaves smart phones or laptops away. Therefore, we decided that if we develop a program named PERTASKWEB that reminds people to what they have to do, then they do not miss anything with this project. In this literature review, some research has been made about personal task managers and previous developed task manager programs and mind map concept.

We discover that there are some programs with different properties to serve for people. But they have some lack of user needs and we plan to fix this problem with our program PERTASKWEB. Our program is web based and in our native language, Turkish. Language support for Turkish of web based task organizing software is rare and only available in few programs, such as Evernote, Todoist and Flow. So it is a big opportunity for people who want to invest money or source to such projects with Turkish language support. In addition, according to our research about mind mapping concept, although graphical representation are not easy to integrate into our program, we added some properties of mind mapping such as treeview of categories.

We also added some parts of our meeting, that done by between our group and Mr Akçalı who is one of the project managers in Havelsan, contents. This meeting's aim was to learn something about mind map concept from a professional business man.

There are a number of documents which explain the methods for task organization or task management. One of these methods is *Kanban* developed in the 1940s by Toyota, a Japanese company. The word *Kanban* means “sign” or “card” in Japanese language. In this method, a white board is divided it into several columns, such as “not started”, “in progress”, “waiting approval” and “done”. Then all of the tasks are put on post-it notes in the left-hand column. As the users start doing tasks or make progress on them, they move the corresponding post-it notes from one column to the next column. Kanban is a way to track and visualize the tasks as



described in *Brandal (2018)*. Additionally, *Brandal (2018)* also evaluates the pros & cons of some tools for task management.

When we started this project, we observed that the subject *Personal Task Management* has attracted the attention of many researchers for a long time starting from 2004. *Belotti (2004)* and others investigated 595 resources on to-do lists and discovered 228 unique tasks whereas 45 of them were carried over one or more weeks. They developed a simple desktop application called as *TaskVista* which lists, sorts and filters the to-do tasks on a single page.

MSc thesis of *Botla (2013)* is a more comprehensive literature work on personal task management. This thesis explains the design of personal assistant software for task management using web and database technologies. This document is very detailed for our project, but it may be very useful for us and for those who plan MSc education on this subject.

Chasanidou,D and others (2013) focuses on the importance of the task management tools for team collaboration. They state that a web based task management system will dramatically increase the cooperation capability of the members of a project.

2.1 Existing Desktop Software Tools for Task Management

There are a number of software tools for personal task management. In order to be acquainted with personal task management, the following software tools are downloaded, installed and compared:

- Task Coach (Free)
- Task Unifier (Free)
- Swift To-Do List (Trial for 30 days)
- Leader Task (Trial for 7 days)
- My Life Organized (Trial for 45 days)
- EverNote (Limited Version)



2.1.1 Task Coach

Task Coach is a free desktop application and can be downloaded from <http://www.taskcoach.org/>.

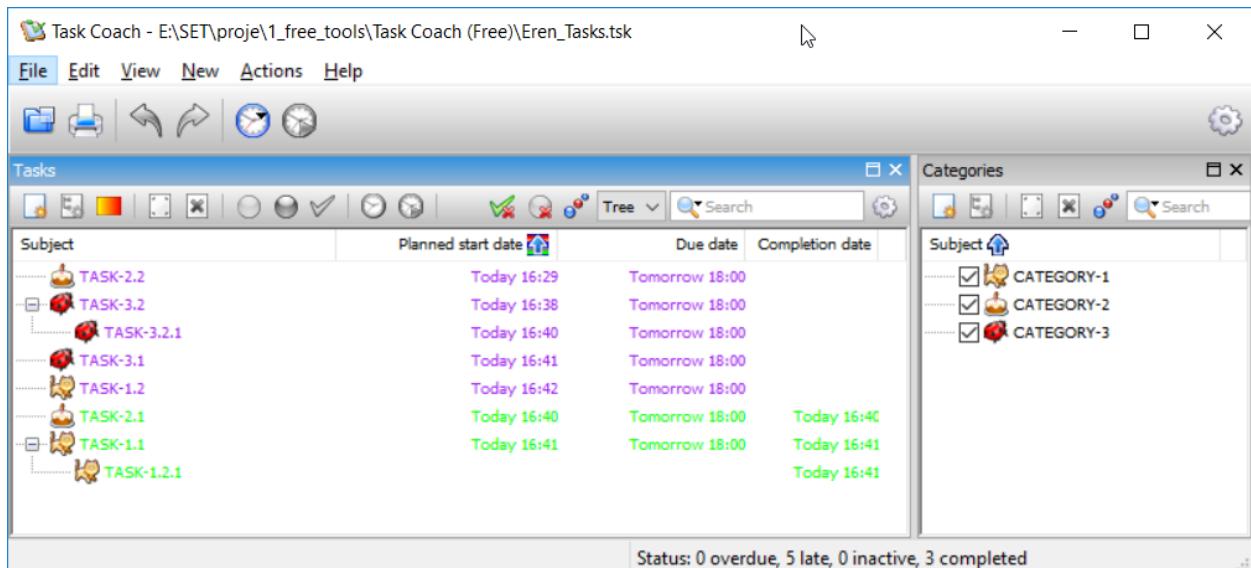


Figure 2.1 Task Coach User Interface

2.1.2 Task Unifier

It is a free desktop application. It can be downloaded from <http://taskunifier.sourceforge.net/>.

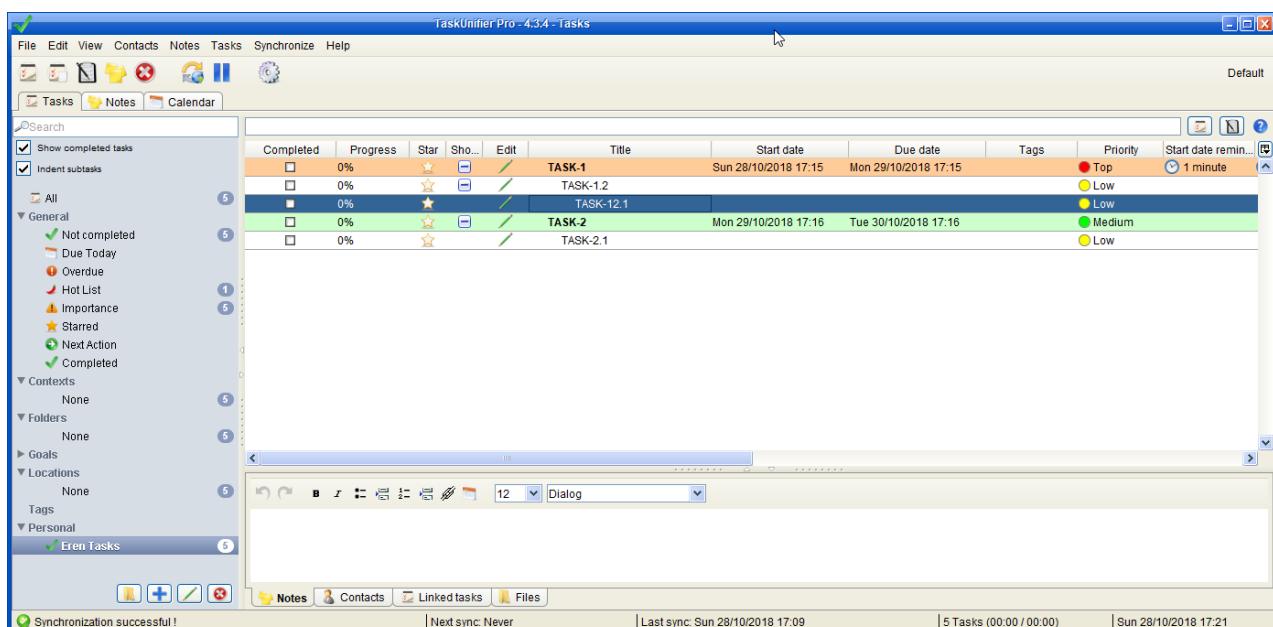


Figure 2.2 Task Unifier User Interface



2.1.3 Swift To-Do-List

It is not a free desktop application. Its costs 99\$. But a 30 day trial version can be downloaded from <http://www.dextronet.com/>.

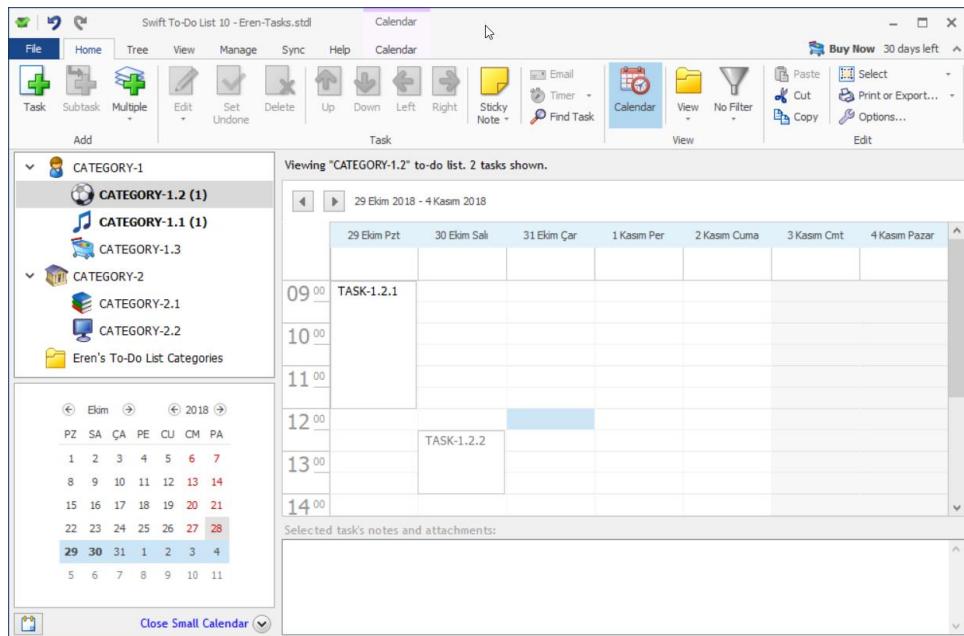


Figure 2.3 Swift To-Do List User Interface

2.1.4 Leader Task

It is not a free application. It has mobile and desktop versions. Its costs 35.94 99\$ for one year and for one user. But a 7 day trial version can be downloaded from

<https://www.leadertask.com/> .

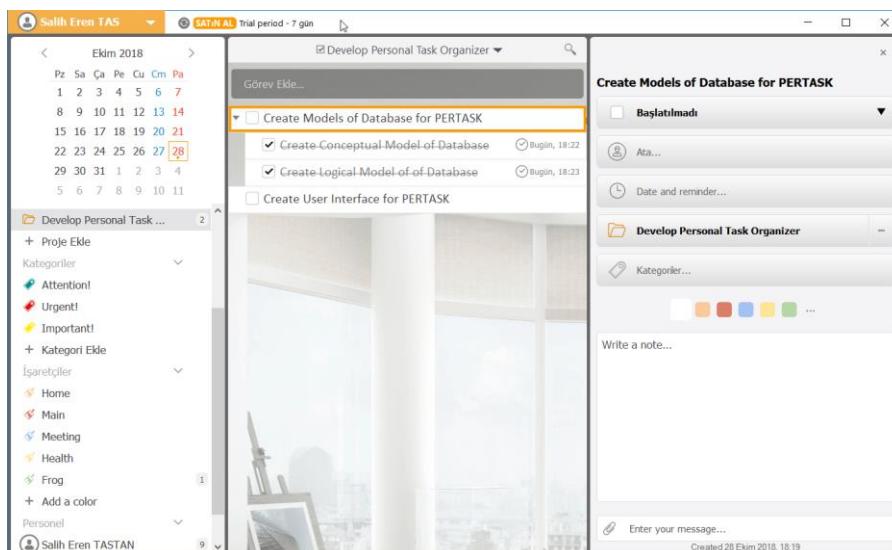


Figure 2.4 Leader Task User Interface



2.1.5 My Life Organized

It is not a free application. It has mobile and desktop versions. Its costs 59.95\$. But a 45 day trial version can be downloaded from <https://www.mylifeorganized.net/>.

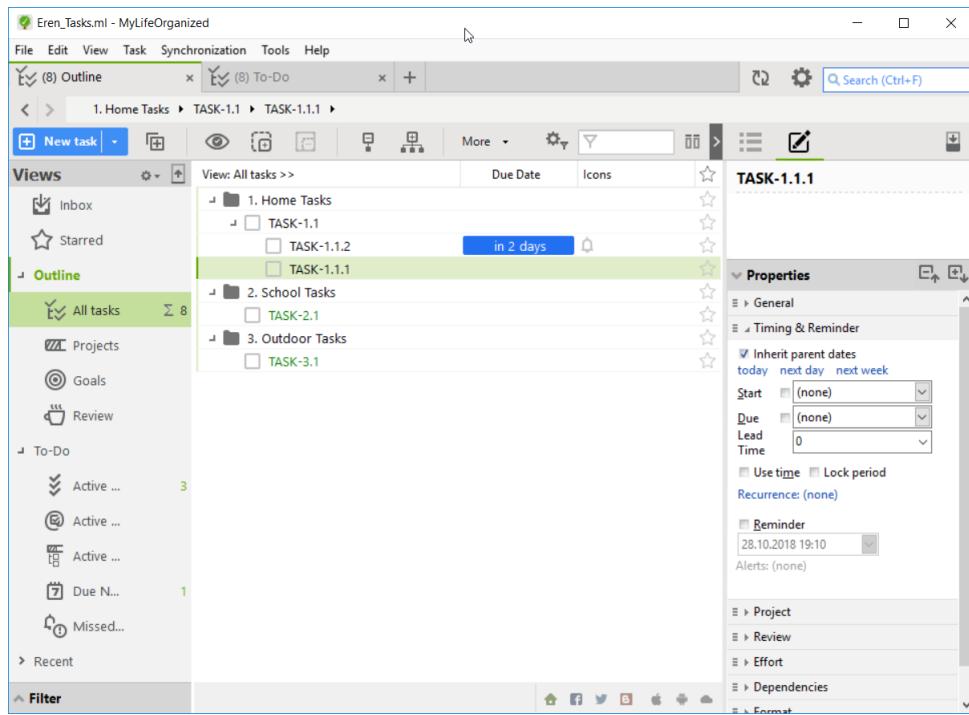


Figure 2.5 My Life Organized User Interface

2.1.6 Ever Note

It is free for basic (limited) version. It has mobile and desktop versions. The costs of premium version and business version are 8.49 TRY and 22 TRY for one month for one user, respectively. Basic version can be downloaded from <https://evernote.com/intl/en/download>.

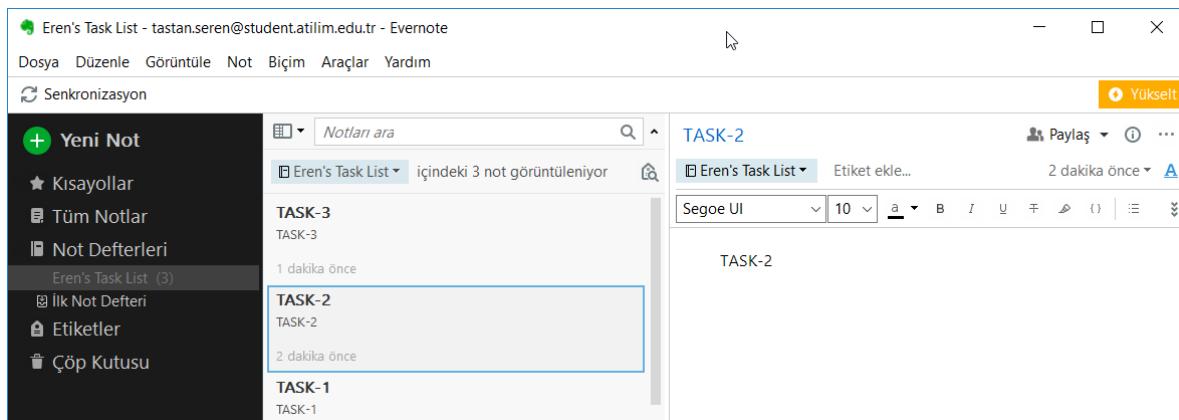


Figure 2.6 Ever Note User Interface



2.1.7 Comparison of Desktop Software Tools for Task Management

Existing desktop software tools for task management are downloaded, installed and compared using various criteria. Although they all have (naturally) add-delete-edit task functionalities, all of them do not support treeview and calendar view capabilities. Only two of them (Task-unifier, task-coach) are totally free of charge. The comparison results are listed in Table 2.1.

Table 2.1 Existing desktop software tools and their capabilities

Functions		Task Coach	Task Unifier	Swift To-Do List	Leader Task	MyLife Organized	Ever-Note
Task	Add	YES	YES	YES	YES	YES	YES
	Delete	YES	YES	YES	YES	YES	YES
	Edit	YES	YES	YES	YES	YES	YES
	Priority	YES	YES	YES	YES	YES	NO
	Description	YES	YES	NO	YES	YES	NO
	Note	YES	YES	YES	YES	NO	NO
	Sub Task	YES	YES	YES	YES	YES	NO
	Schedule	YES	YES	YES	YES	YES	NO
	Unscheduled	YES	YES	YES	YES	YES	NO
Task List	Expand Tree	YES	NO	YES	YES	YES	NO
	Collapse Tree	YES	NO	YES	YES	YES	NO
	Save (Export)	YES	YES	YES	NO *	YES	NO *
	Open (Import)	YES	YES	YES	NO *	YES	NO *
Tasks	Export (txt, csv)	YES	NO	NO	NO	YES	NO
	Import (txt, csv)	YES	NO	YES	NO	YES	NO
Category (List, Project)	Add	YES	NO	YES	YES	YES	YES
	Delete	YES	YES	YES	YES	YES	YES



Calendar View of Tasks	Day	NO	YES	YES	YES	NO	NO
	Week	NO	YES	YES	NO	NO	NO
	Month	NO	YES	YES	NO	NO	NO
Date	Start	YES	YES	YES	YES	YES	NO
	Due	YES	YES	YES	NO	YES	NO
	Completion	YES	YES	YES	NO	YES	NO
Sort by	Creation Date	YES	YES	NO	NO	YES	YES
	Completion Date	YES	YES	NO	NO	YES	NO
	Priority	YES	YES	NO	YES	YES	NO
Show only	Completed	YES	YES	YES	NO	YES	NO
	Not completed	NO	YES	YES	NO	YES	NO
	Due today	NO	YES	YES	NO	NO	NO
	Overdue	NO	YES	YES	NO	NO	NO
	Unscheduled	YES	YES	YES	NO	NO	NO
Reminder	At a time + date	YES	YES	YES	YES	YES	NO
	On start date	NO	NO	YES	NO	NO	NO
	On due date	NO	NO	YES	NO	NO	NO
	... minutes before start date	NO	YES	YES	NO	NO	NO
	... minutes before due date	NO	YES	YES	NO	NO	NO
Turkish Language Support		YES	NO	YES	NO	NO	YES
Help		YES	YES	YES	YES	YES	YES
Free		YES	YES	NO	NO	NO	YES



(Trial period)				(30 days)	(7 days)	(45 days)	(Basic)
Cost (valid for)		0	0	99 \$ (forever)	36 \$ (1 year)	60 \$ (forever)	0

(*) open & save with user & password

2.1.8 Comparison of Web-based Software Tools for Task Management

As we have done for desktop tools, we also compared existing web-based software tools for task management using the same criteria . As compared to desktop software, they have more functionalities but few language support for Turkish but two, namely Todoist and Flow. They all have , calendar view capability but not treeview. Three of six (Todoist, Toodledo and TickTick) are free of charge. The detailed comparison results are listed in Table 2.2.

Table 2.1 Existing Web-based software tools and their capabilities

Functions		Todoist	Toodledo	TickTick	Doit.im	Week Plan	Microsoft Flow
Task	Add	YES	YES	YES	YES	YES	YES
Task	Delete	YES	YES	YES	YES	YES	YES
	Edit	YES	YES	YES	YES	YES	YES
	Priority	YES	YES	YES	YES	YES	YES
	Description	YES	YES	YES	YES	YES	YES
	Note	YES	YES	YES	YES	YES	YES
	Sub Task	YES	YES	YES	YES	YES	YES
	Schedule	YES	YES	YES	YES	YES	YES
	Unscheduled	NO	YES	YES	YES	YES	YES
Task List	Expand Tree	NO	NO	YES	NO	NO	YES



	Collapse Tree	NO	NO	YES	NO	NO	YES
	Save(Export)	YES	YES	NO	YES	YES	NO
	Open(Import)	YES	YES	NO	YES	YES	NO
Tasks	Export	YES	YES	NO	YES	YES	NO
	Import	YES	YES	NO	YES	YES	NO
Category (List, Project)	Add	YES	YES	YES	YES	NO	YES
	Delete	YES	YES	YES	YES	NO	YES
Calendar View of Tasks	Day	YES	YES	YES	YES	YES	YES
	Week	YES	YES	YES	YES	YES	YES
	Month	NO	YES	YES	YES	YES	NO
Date	Start	YES	YES	YES	YES	YES	YES
	Due	YES	YES	YES	YES	YES	YES
	Completion	YES	YES	YES	YES	YES	YES
Sort by	Creation date	YES	YES	YES	YES	YES	YES
	Completion date	YES	YES	YES	YES	YES	YES
	Priority	YES	YES	YES	YES	YES	YES
Show Only	Completed	YES	YES	YES	YES	YES	YES
	Not completed	YES	YES	YES	YES	YES	YES



	Due today	YES	YES	YES	YES	YES	YES
	Overdue	NO	YES	YES	NO	NO	NO
	Unscheduled	NO	YES	YES	NO	NO	NO
Reminder	At a time + date	YES	YES	YES	YES	YES	NO
	On start date	YES	YES	YES	YES	YES	NO
	On due date	YES	YES	YES	YES	YES	NO
 minutes before start date	NO	YES	YES	YES	NO	NO
 minutes before due date	YES	YES	YES	YES	NO	NO
Help		YES	YES	YES	YES	YES	NO
Turkish Language Support		YES	NO	NO	NO	NO	YES
Free (Trial Period)		YES	YES	YES	NO (30 days)	YES (Basic)	NO
Cost (valid for)		0	0	0	\$2.00 per month	0	\$4.79 per month



3. Requirements

3.1 Overall Description of the Project

3.1.1 Product Perspective

PERTASKWEB will be a web based software suitable for all users interested in organizing and cooperating their personal tasks as well as project tasks. This software will be composed of a set of PHP files and will be installed on a WAMP (Windows, Apache, MySQL, PHP) server (<http://www.wampserver.com/en/>) which operates on Windows and can be used via any web browser.

3.1.2 Product Functions

General functions of PERTASKWEB are:

- Creating new tasks and subtasks.
- Editing existing tasks and their attributes, like notes, creation time, priority, due date.
- Creating new categories of tasks (task groups, such as Home Tasks, School Tasks, etc.)
- Editing existing categories (task groups)
- Deleting tasks and categories (task groups)
- Move tasks between categories (task groups)
- Sorting tasks by their attributes.
- Mark a task as completed.
- Save all tasks in a database (MySQL).
- Using on the web site which will be hosted at a server (localhost or remote host).
- Adding- Deleting-Editing Users.
- Using with user and password by many users with a login page.
- Adding-Deleting-Editing User Groups (such students, teachers, etc.)
- Assigning Users to one or more User Groups.
- Adding-Deleting-Editing Projects containing Tasks and Sub-tasks.

Specific functions of PERTASKWEB are:



- PERTASKWEB will have 3 working areas, which can be closed and opened independently:
 1. Calendar area, which can have day, week, month, or year view showing projects, tasks and sub tasks
 2. Daily to-do list, things planned for that day but not scheduled.
 3. Hierarchical project/task organization of one's all aspects of life.
 4. Reaching all members of the system (users, user groups, projects, tasks, sub-tasks) from any web page
- The user can carry any item from the 3rd area to the 1st area if it can be scheduled for a particular date/time or to the 2nd area if it is planned for the current date but doesn't have a particular time. The calendar area can contain items that have a specific date but no scheduled time. These items will be automatically carried onto the 2nd area (daily to-do list) when that date comes.

3.1.3 User Characteristics

Users are not required to have a specialized instructional and academic background but some basic knowledge of internet use.

3.1.4 Constraints

PERTASKWEB can be installed on any PHP supported Web Server and can be used from any web browser at desktop, laptop and tablet computers as well as mobile phones with an Internet connection.

3.1.5 Assumptions and Dependencies

There are no conditions for using the PERTASWEB but a need to Internet connection.

3.2 Specific Requirements

3.2.1 Main

3.2.1.1 Minimize to tray

Description: In this function, it is described how user can minimize PERTASKWEB, into a software tray on the operating software's taskbar.



The user: User chooses "Web Browser's Minimize Icon".

The software: When user chooses "Main: Minimize to tray", software must hide the open window of PERTASKWEB and create a small tray icon on the operating software's taskbar.

Constraints: There are not any constraints.

3.2.1.2 Exit

Description: In this function, it is described, how users can exit the program.

The user: User chooses "View: Exit".

The software: When user chooses "View: Exit", software must save anything has not being saved yet and then terminate the program.

Constraints: There are not any constraints.

3.2.2 Category

In this menu there are many functions that must show up when user right clicks on a category. Functions in this category are about creating, editing and deleting categories.

3.2.2.1 New category

Description: In this function, it is described how users can create a new category.

The user: User has to choose "Category: New category" and then a pop up window shows up, with a form, in which user must complete the category name. After completing the fields, user must press add.

The software: The software must show up a pop up window, when the user chooses "Category: New category". When the user presses add, software must check if the "name" field is completed. If it is not, it should warn the user that this field is mandatory, with a new pop up box, and allow the user to continue completing in the first pop up window. Finally, the software must save the new category and its attributes.

Constraints: A new category cannot be added if the "name" field is not completed.

3.2.2.2 Edit category

Description: In this function, it is described how users can edit an existing category.

The user: Firstly, user must select an existing category by left clicking on it and then choose "Category: Edit category". Then a pop up window will show up, same one as the new



category's, in which the user can edit the category's fields. User must not delete the name of the category and leave it empty, as it is mandatory.

The software: Software must show up a pop up window, when the user chooses "Category: Edit category". When the user presses edit, software must check if the "name" field is completed. If it is not, it should warn the user that this field is mandatory, with a new pop up box, and allow the user to continue completing in the first pop up window. Also software, must not allow the user to be able to press "Category: Edit category", if a category has not been already selected. Finally, the software must save the edited category and its attributes.

Constraints: Edited category cannot be saved if the new name of it is empty and if there is not a category selected before user chooses edit category.

3.2.2.3 Delete category

Description: In this function, it is described how the user can delete a category.

The user: User selects a category. Then chooses "Category: Delete category" and a dialog box shows up, asking if he wants to delete the selected category. If user answers positive, then this category and any task in it, will be deleted, else the delete command will be canceled.

The software: When user chooses "Category: Delete category", software shows up a dialog box asking the user, if he wants to delete the selected item. If user answers positive, software deletes the selected category, and any other task there is in it, else software cancels this action.

Constraints: There must be a selected category before user chooses delete category.

3.2.3 Task

Functions in this category are about creating, editing and managing tasks.

3.2.3.1 New task

Description: In this function, it is described the procedure of creating a new task, completing task's fields and adding the task in an existing category. Each task will be a parent node just like in mind maps. (See **Appendix: Interview with Mr Akın at HAVELSAN**)

The user: User has to choose "Task: New task" and then a pop up window will shows up, with the empty fields of the task and the category, in which the task will be added in. If there is not already a category, user should create a new one by pressing new category. User must definitely complete the field with the name of the task, complete the description, set the priority of the



task, define the due date , set the task as completed, set the task timer and add a note (if needed)

The software: Software must show up a pop up window, when the user chooses ‘Task: New task’. When the user presses add, software must check if the "name" field is completed. If it is not, it should warn the user that this field is mandatory, with a new pop up box, and allow the user to continue completing in the first pop up window. Also software, must not allow the user to add a new task if there is not an available category. Finally, the software must save the new task and its attributes, and then add it nested in the category.

Constraints: A new task cannot be added if the "name" field is not completed and if there is not a category to add the new task.

3.2.3.2 New subtask

Description: In this function, it is described the creation of a new subtask, which will be added in an existing task, and the completion of its fields. Each subtask will be a child node of the parent node, just like the tree in mind map. The task will be starting with a parent node. (See **Appendix:** Interview with Mr Akin at HAVELSAN)

The user: Firstly, the user will select an existing task by clicking on it and then choose "Task: New subtask of selected one". Then a pop up window will show up, almost same one as the new task's, in which the user can complete the subtask's fields. User must complete the name of the task, as it is mandatory, and complete the description, set the priority of the task, define the due date, set the task as completed, set the task timer, and add a note.

The software: The software must show up a pop up window, when the user chooses "Task: New subtask of selected one". This pop up window is almost the same as the new task's pop up window, with the only difference, that it has not the category field. When the user presses add, software must check if the "name" field is completed. If it is not, it should warn the user that this field is mandatory, with a new pop up box, and allow the user to continue completing in the first pop up window. Also software, must not allow the user to be able to press "Task: New subtask of selected one", if a task has not been already selected. Finally, the software must save the new task and its attributes, and then add it nested in the parent task.

Constraints: A new task cannot be added if the "name" field is not completed and if there is not a task selected, so that the new subtask can be put nested in it.



3.2.3.4 Edit Task

Description: In this function, it is explained the procedure of editing an existing task and its fields.

The user: Firstly, user must select an existing task by left clicking on it and then choose "Task: Edit task". Then a pop up window will show up, same one as the new task's, in which the user can edit task's fields. User must not delete the name of the task and leave it empty, as it is mandatory. User can also change the color, in which the task will be shown, edit the description, reset the priority of the task, redefine the due date, set the task as completed, edit the task timer, reset the task timer, sum the subtasks' timers, edit payment details, add, edit and delete a note.

The software: Software must show up a pop up window, when the user chooses 'Task: Edit task'. When the user presses edit, software must check if the "name" field is completed. If it is not, it should warn the user that this field is mandatory, with a new pop up box, and allow the user to continue completing in the first pop up window. If user presses sum subtasks, software must sum the timers of each subtask of the current task and return the result in the current task's timer. Also software, must not allow the user to be able to press "Task: Edit task", if a task has not been already selected. Finally, the software must save the edited task and its edited attributes, and move it in a new category or parent task, if the user has changed the task's location.

Constraints: Edited task cannot be saved if the new name of it is empty and if there is not a task selected before user chooses edit task.

3.2.3.5 Move task

Description: In this function, it is described how the user can move a task or a subtask, from a category or a parent task to other categories or parent tasks. In other words, the user will change the parent node of a child node just like in mind maps (See **Appendix: Interview with Mr Akin at HAVELSAN**)

The user: First of all, the user selects a task or a subtask. Then chooses "Task: Move task" and a new pointer shows up. The user left clicks with the new pointer on a category or a task that he wishes the selected task to move in.

The software: When the user chooses "Task: Move task", the software changes the pointer of the mouse to show to the user that he is going to move a task. When, the user has left clicked on a category or a parent task, the software must move the selected task from its former location



to the new location user has chosen. Software must save the new location of the selected task.

Constraints: In order to move a task, there must be at least one different category or task to move to and a task must be selected before user chooses move task.

3.2.3.6 Delete task

Description: In this function, it is described how the user can delete a task or a subtask. In other words, the user will delete the parent node or a child node and if a parent node deleted then all child nodes of this parent node will also be deleted, just like in mind maps (See **Appendix: Interview with Mr Akin at HAVELSAN**)

The user: User selects a task or a subtask. Then chooses "Task: Delete task" and a dialog box shows up, asking if he wants to delete the selected task. If user answers positive, then this task and any subtask in it, will be deleted, else the delete command will be canceled.

The software: When user chooses "Task: Delete task", software shows up a dialog box asking the user, if he wants to delete the selected item. If user answers positive, software deletes the selected task, and any other subtask there is in it, else software cancels this action.

Constraints: There must be a selected task before user chooses delete task.

3.2.3.7 Mark task completed

Description: In this function, it is described the procedure of marking a task as completed. Every task must have a checkbox on its left, to make it clear if the task is completed or not.

The user: User selects a task and then chooses 'Task: Mark task completed'. After completing this action, selected task's checkbox should be checked. Alternatively, user can straight check the checkbox by himself.

The software: When user chooses "Task: Mark task completed", the software must check the checkbox of the selected task and save this task as completed.

Constraints: There must be a selected task, which is uncompleted, before user chooses mark task completed.

3.2.3.8 Mark task uncompleted

Description: In this function, it is described the procedure of marking a task as uncompleted. Every task must have a checkbox on its left, to make it clear if the task is completed or not.

The user: User selects a task and then chooses "Task: Mark task uncompleted". After



completing this action, selected task's checkbox should be unchecked. Alternatively, user can straight uncheck the checkbox by himself.

The software: When user chooses "Task: Mark task uncompleted", the software must uncheck the checkbox of the selected task and save this task as uncompleted.

Constraints: There must be a selected task, which is completed, before user chooses mark task uncompleted.

3.2.3.9 Mark task unscheduled

Description: In this function, it is described the procedure of marking a task as unscheduled. Every task must have a due date, to make it clear if the task is scheduled or not.

The user: User selects a task and then chooses "Task: Mark task unscheduled". After completing this action, selected task's due date will be deleted. Alternatively, user can straight define a new due date by himself.

The software: When user chooses "Task: Mark task unscheduled", the software must delete the due date of the selected task and save this task as unscheduled.

Constraints: There must be a selected task, which is scheduled, before user chooses mark task unscheduled.

3.2.4 Calendar

3.2.4.1 Today

Description: In this function, user can see all tasks scheduled for today.

The user: User chooses "Calendar: Today"

The software: The software shows all tasks scheduled for today. Also the software shows tasks scheduled and not scheduled in left side within different boxes.

Constraints: There are not any constraints.

3.2.4.2 Day Calendar

Description: In this function, user can see day calendar with tasks if there are.

The user: User chooses "Calendar: Day."

The software: The software shows day calendar with tasks if there are. Also the software shows tasks scheduled and not scheduled in left side within different boxes.



Constraints: There are not any constraints.

3.2.4.3 Week Calendar

Description: In this function, user can see week calendar with tasks if there are.

The user: User chooses "Calendar: Week".

The software: The software shows week calendar with tasks if there are. Also the software shows tasks scheduled and not scheduled in left side within different boxes.

Constraints: There are not any constraints.

3.2.4.4 Month Calendar

Description: In this function, user can see month calendar with tasks if there are.

The user: User chooses "Calendar: Month".

The software: The software shows month calendar with tasks if there are. Also the software shows tasks scheduled and not scheduled in left side within different boxes.

Constraints: There are not any constraints.

3.2.4.5 Change Task Area

Description: In this function, it is described how users can change the area of a task. The product will have 3 working areas, which can be closed and opened independently: (1) Calendar area, which can have day, week, month, or year view. (2) Daily to-do list, things planned for that day but not scheduled. (3) Hierarchical project/task organization of one's all aspects of life.

The user: User selects a task and then chooses "Calendar: Change Task Area". After this, the user can carry any task from the 3rd area to the 1st area if the task can be scheduled for a particular date/time or to the 2nd area if it is planned for the current date but doesn't have a particular time.

The software: When user chooses 'Calendar: Change Task Area', the software must move the task to the area selected. The calendar area can contain items that have a specific date but no scheduled time. These items will be automatically carried onto the 2nd area (daily to-do list) when that date comes.

Constraints: There must be a selected task, before user chooses "Calendar: Change Task Area".



3.2.5 View

In this menu there are functions, which change the view of the tasks.

3.2.5.1 Flat View

Description: In flat view, every task and subtask of each category, show up in the same list allowing the user to handle them together. For example, instead of only sorting the top tasks, user can also sort the subtasks at the same time.

The user: User chooses "View: Flat view".

The software: When user chooses "View: Flat view", software must unravel the nested subtasks, and show every task and their subtasks in the same list. When user tries to sort the tasks by any attribute, software must also sort the whole list, including the subtasks.

Constraints: There are not any constraints.

3.2.5.2 Expand all Tasks

Description: In this function, every task, which has at least one subtask, expands and shows its nested subtasks. In other words, the user will expand all child nodes if there exists at least one child of a parent, just like in mind maps (See **Appendix:** Interview with Mr Akin at HAVELSAN).

The user: User chooses "View: Expand all".

The software: When user chooses "View: Expand all", software must show the nested subtasks of each task. However, if user tries to sort the tasks by any attribute, software will only sort the top tasks.

Constraints: There are not any constraints.

3.2.5.3 Collapse all Tasks

Description: In this function, every expanded task, which has at least one subtask, must hide its own subtasks. It is the exact opposite function of Expand all. In other words, the user will collapse all child nodes if there exists at least one child of a parent, just like in mind maps (See **Appendix:** Interview with Dr. Akın Kağan AKÇALI at HAVELSAN)

The user: User chooses "View: Collapse all".

The software: When user chooses "View: Collapse all", software must hide the nested subtasks of each task.



Constraints: There are not any constraints.

3.2.5.4 Show completed Tasks

Description: In this function, it is described, how users can show the completed tasks.

The user: User chooses "View: Show completed tasks".

The software: When user chooses "View: Show completed tasks", software must show any task that is completed in the task list. User must not be able to choose show completed tasks, if these tasks are already shown.

Constraints: Completed tasks must be hidden, so that user can choose show completed tasks.

3.2.5.5 Hide completed Tasks

Description: In this function, it is described, how users can hide the completed tasks.

The user: User chooses "View: Hide completed tasks".

The software: When user chooses "View: Hide completed tasks", software must hide any task that is completed from task list. User must not be able to choose hide completed tasks, if these tasks are already hidden.

Constraints: Completed tasks must be shown, so that user can choose hide completed tasks.

3.2.6 Register, Login and Log Out

Here explained how user register, log in and log out, and software and user responsibilities are descripted.

3.2.6.1 Register

Description: In this function, it is described that how user register to system.

The User: User opens the registration page and specifies his information. The username and password information should be entered.

The Software: The system should validate any of the username and password information are not belong to someone else. If it is, then system should warn to user to entered whichever belongs to someone else. If it is not, then system should register the user to system.

Constraints: The password and username information should not be entered before by someone else.



3.2.6.2 Login

Description: In this function, it is described that how user log in to system

The User: User opens the login page and writes the username and password into the related boxes then press the login button. After system validate the specified information, user logs in to system.

The Software: The system should check the entered information after the user pressed login button. If any of the input boxes are empty, system should warn the user. If system validate the entered information then should enter the user to system.

Constraints: The user must have registered first. The entered information should be validated by the system.

3.2.6.3 Logout

Description: In this function, it is described that how user log out to system

The User: User clicks the logout button at the top right of the page.

The Software: The system logs out the user and directs him to login page after user pressed the logout button at the top right of the page.

Constraints: There is no constraint to log out from the system.

3.2.7 Implementation Results of the Functional requirements

All the requirements are not implemented. Some of them implemented partially depending upon the unimplemented requirements. And some extra requirements needed are implemented. The main reason for unimplemented requirements was the lack of time for coding. As a result 77 % of the total requirements are implemented. (See Table 3.1)

Table 3.1 Summary of the Implementation Results of the Functional Requirements

TOTAL NO OF REQUIREMENTS	30
NO OF IMPLEMENTED	23
NO OF NOT IMPLEMENTED	7
PERCENTAGE OF IMPLEMENTED	77%

The details of the implementation results of the functional requirements are listed in Table 3.2.

**Table 3.2** Implementation Results of the Functional Requirements

Functional Requirements		Implemented	Remark/ Why Not implemented?
Main	Minimize to Tray	YES	Browser Minimize
	Exit	YES	Logout option
Category	New Category	YES	
	Edit Category	YES	
	Delete Category	YES	
Task	New Task	YES	
	Edit Task	YES	
	Move Task (to another category)	YES	
	Delete Task	YES	
	New Sub-Task	NO	Lack of time
	Mark Task as Completed	NO	
	Mark Task as Uncompleted	NO	
	Mark Task as Unscheduled	NO	
Calendar	Today	YES	
	Day Calendar	YES	
	Week Calendar	YES	
	Month Calendar	YES	
	Change Task Area	NO	Lack of time
View	Flat View	YES	as Agenda View
	Expand all tasks	YES (partial)	for categories only since no sub-tasks
	Collapse all tasks	YES (partial)	
	Show completed tasks	NO	Lack of time
	Hide completed tasks	NO	
Register,Login,Logout	Register	YES	
	Login	YES	
	Logout	YES	
Extra	Forgot Password	YES	Password is sent to user's email
	Verification Code	YES	Verification code is sent to user's email
	Change Password	YES	Password is changed by the user
	on-line help (html)	YES	



3.3 Non-functional Requirements

3.3.1 Database Requirements

PERTASKWEB requires SQLite database. Database and tables will be created with DDL (create database, create table, add column, etc.) automatically during setup and manipulated with DML (select, insert, update, delete, etc.) within the system as long as it is used by the users.

According to the software requirements, **conceptual model of database** (ER Diagram) for PERTASKWEB software is prepared by using free online tool at <https://www.draw.io/> as shown in the figure on the next page.

According to conceptual model on the next page, **logical model of database** is as follows:

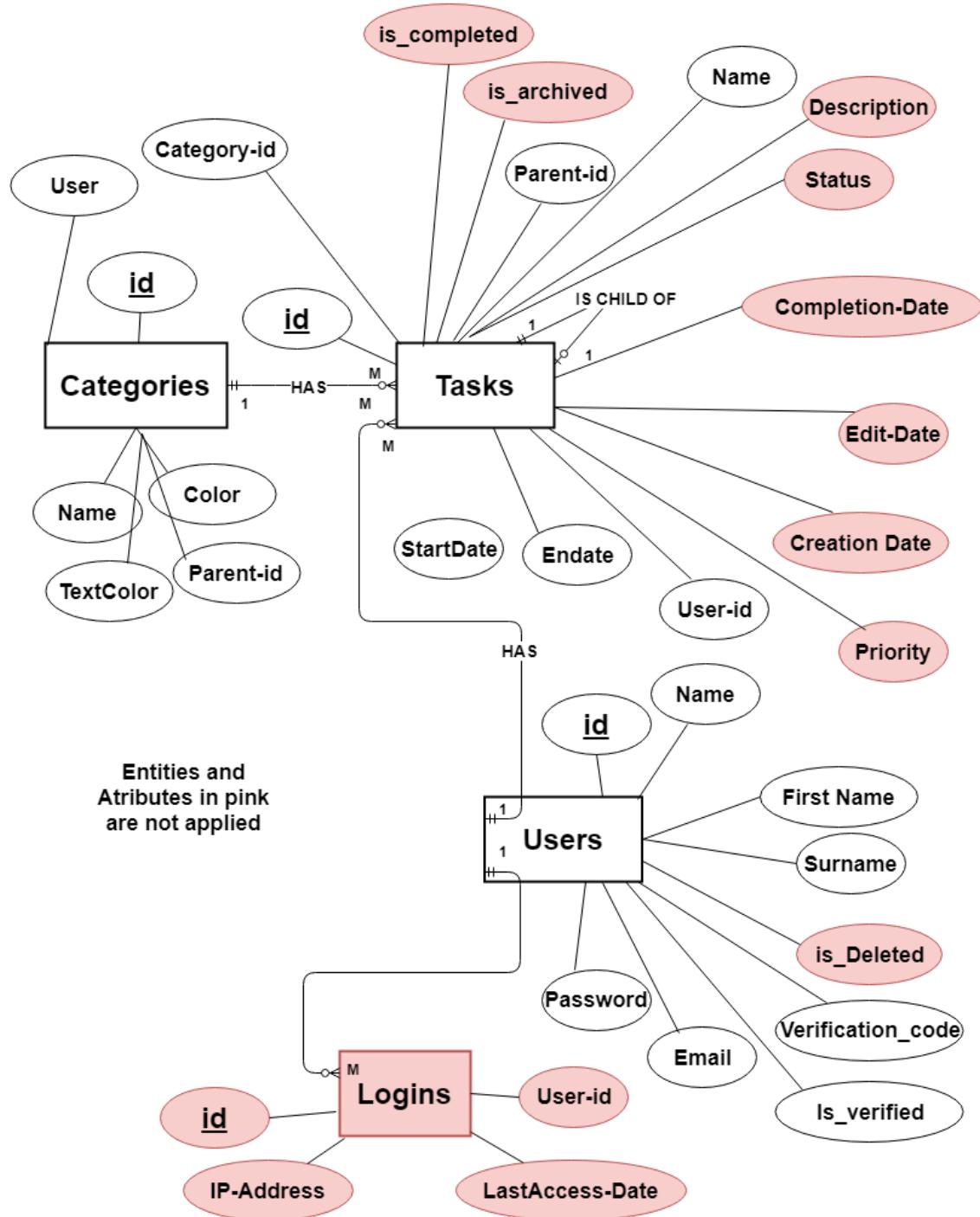
Tasks (id, name, description, status, completed, archived, parentid, completiondate, startdate, enddate, editdate, creationdate, priority, userid) [userid refer to **id** in table Users, , categoryid refer to **id** in table Categories]

Categories (id, name, user, color, textcolor, parentid)

Users (id, name, firstbname, surname, password, email, verificationcode, isverified, isdeleted)

Logins (id, ip, lastaccessdate, userid)) [userid refers to **id** in table Users]

ER Diagram of PERTASKWEB Database



Entities and
Atributes in pink
are not applied



3.3.2 Hardware Interface Requirements

a PC **and an Internet Connection** is required.

3.3.3 Software Interface Requirements

In order to work properly, PERTASKWEB uses WAMP (Windows, Apache, MySQL, PHP) server which is supposed to be installed on Windows Operating System.

3.3.4 Communications Interface Requirements

PERTASKWEB needs a connection with an internet provider and a domain or subdomain (e.g. pertask.atilim.edu.tr) to host server (if not localhost: 127.0.0.1).

3.3.5 Performance Requirements

PERTASKWEB is a program with minimal memory requirements, disk space and processing power.



4. Design

4.1 Software Methodology

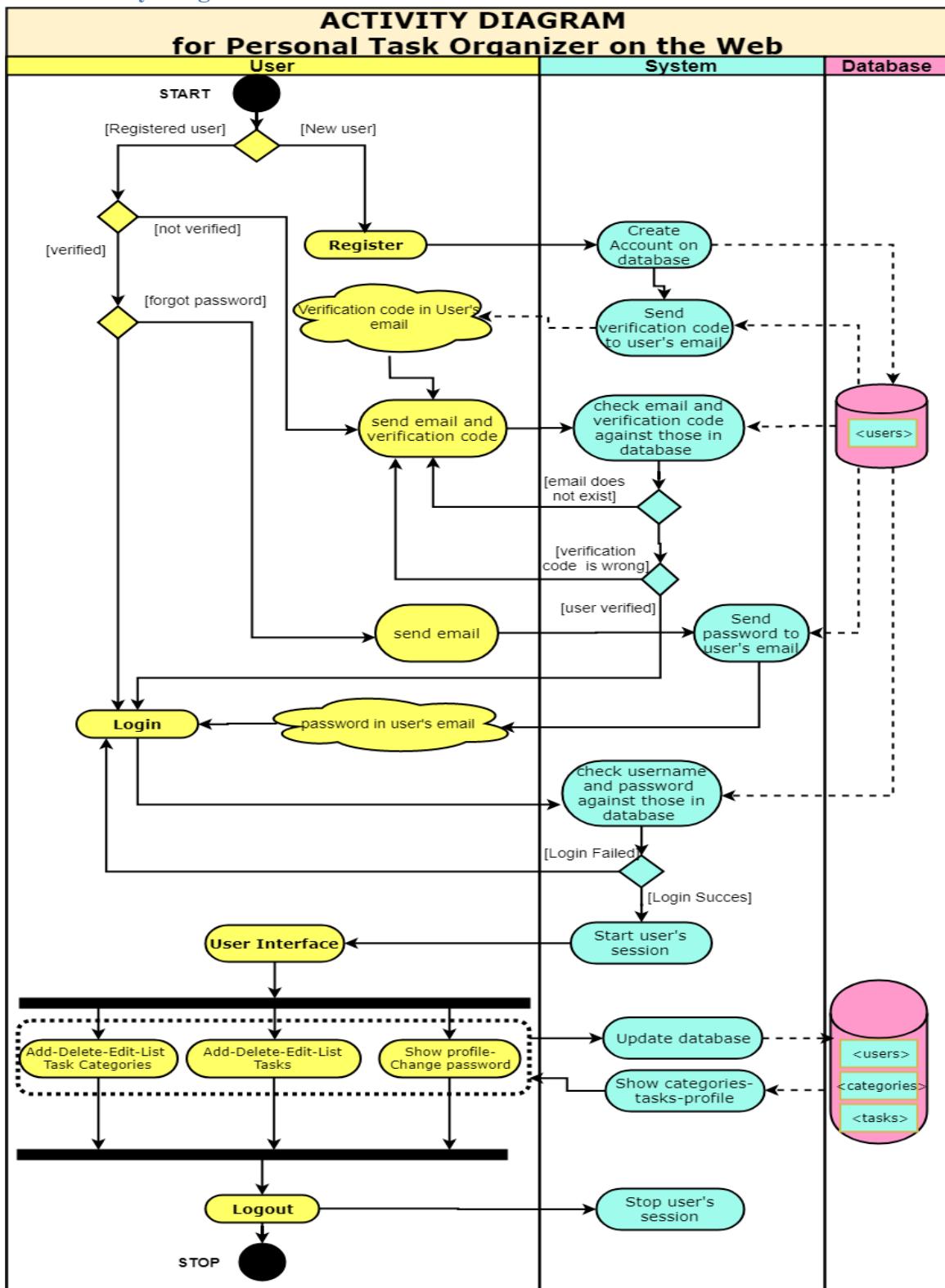
In this project we have used agile software development methodology because of the main advantages of this methodology which are speed, short cycle output generation and feedbacks on thoughts. At the initial phase our objective was to develop a desktop software. But after discussion with our supervisor and investigating the future of technology, we moved to a web based system.

4.2 System Architecture

We designed our system as a web based system. The server side is composed of a PHP enabled Web Server (e.g. WAMP) without any need to a Database Management System since we planned to use SQLite which can be easily used as PDO (PHP Data Object). The client side is planned as an ordinary web browser running at any computer from desktop to mobile phones.

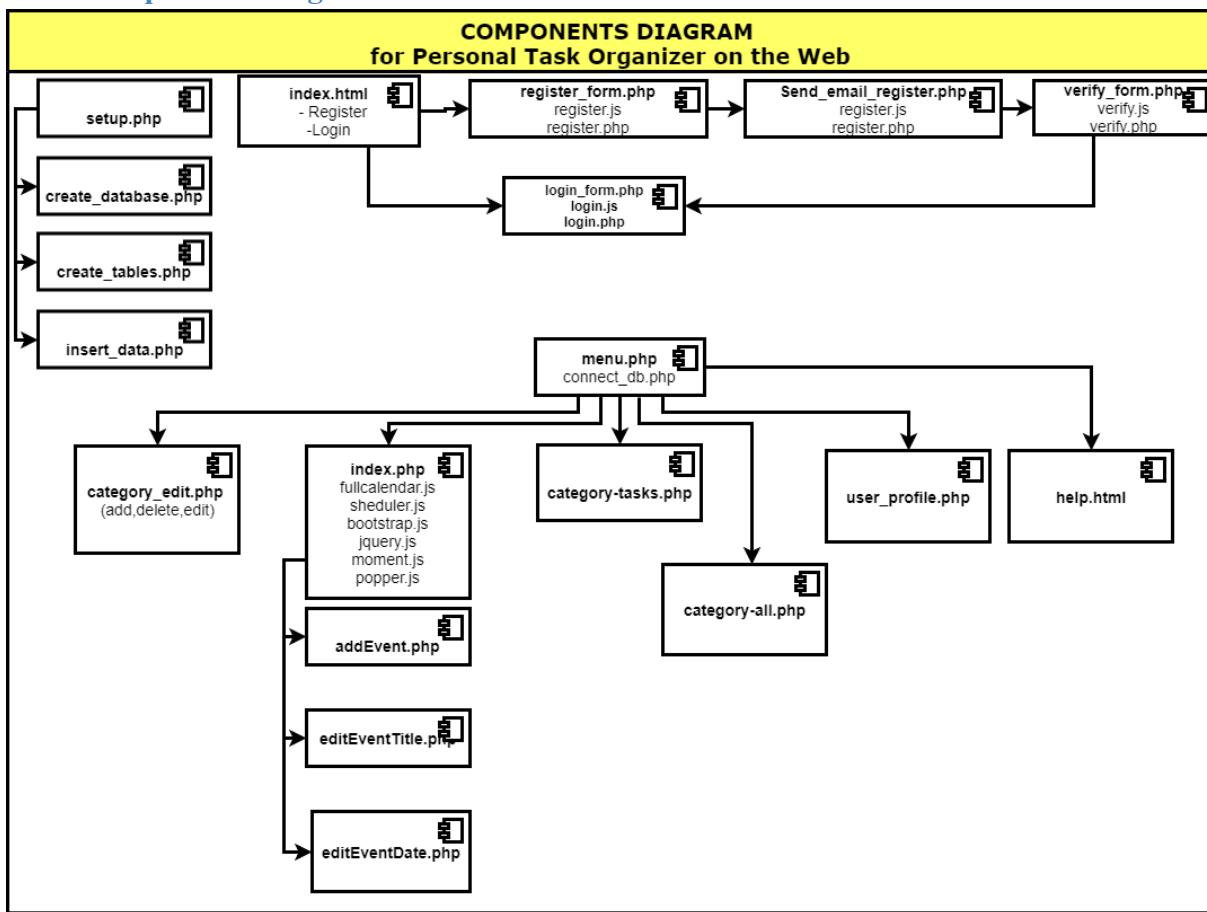
4.3 UML Diagrams of the System

4.3.1 Activity Diagram

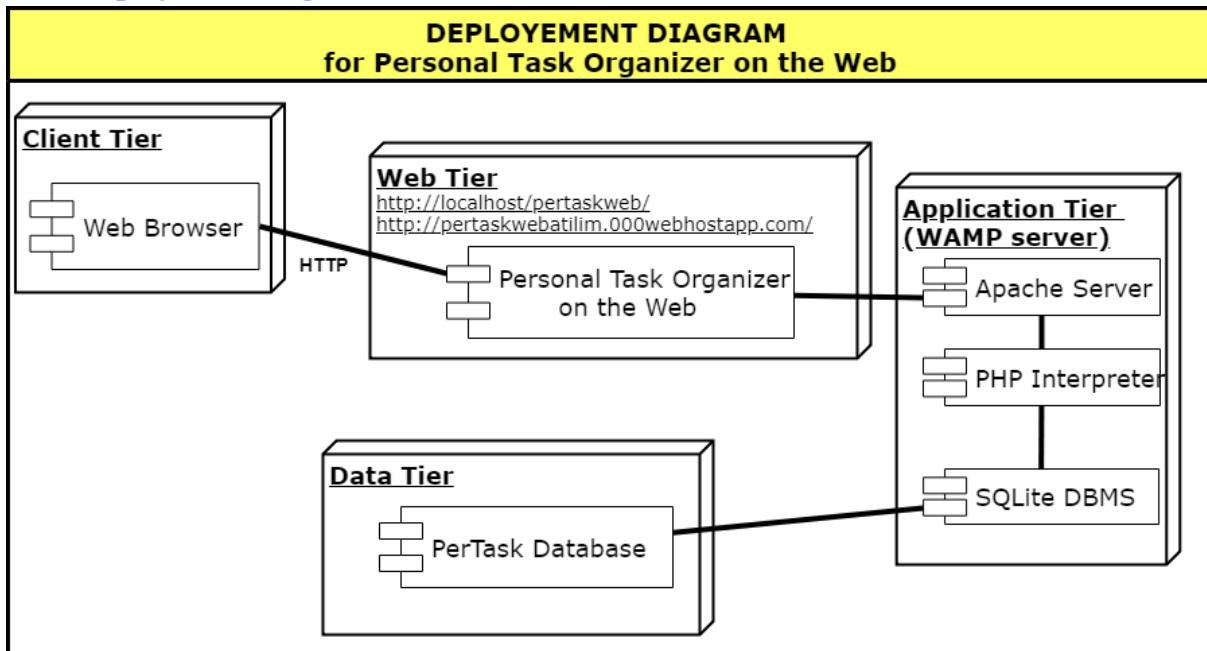




4.3.2 Components Diagram

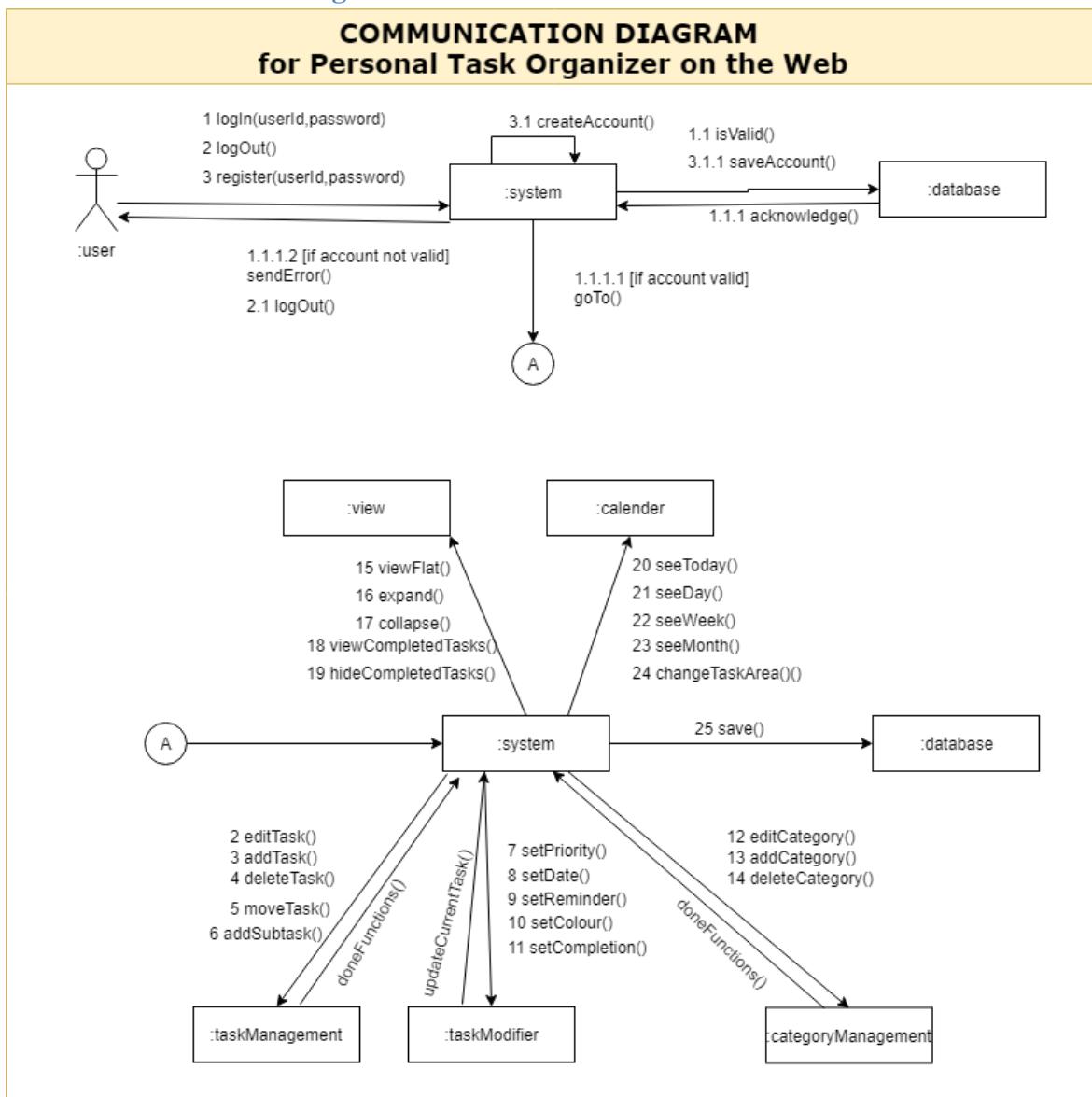


4.3.3 Deployment Diagram



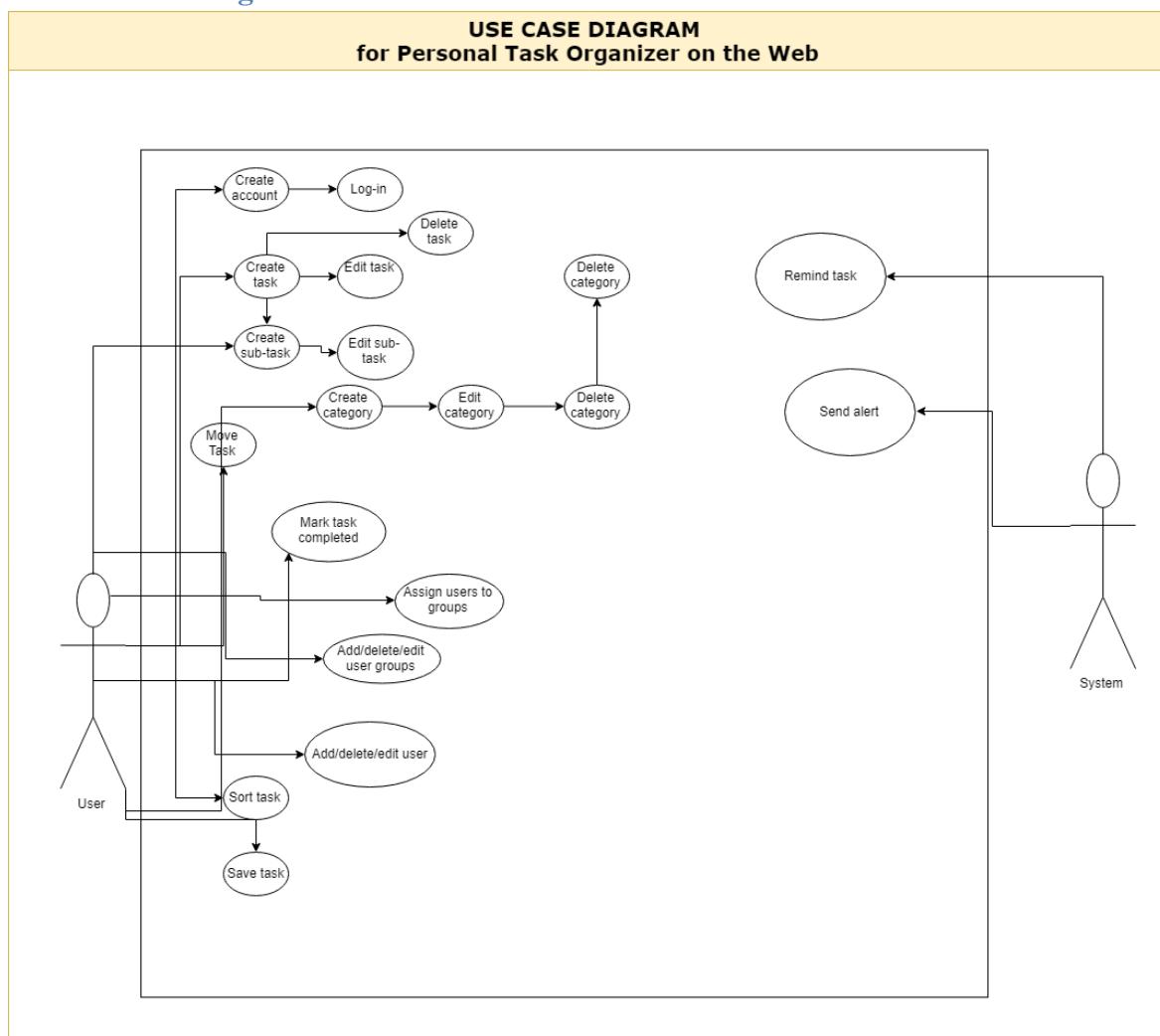


4.3.4 Communication Diagram





4.3.5 Use Case Diagram





4.4 User Interface Design of the System

4.4.1 Login and Register Pages

Personal Task Organizer on the Web - ATILIM UNIVERSITY
http://pertask.atilim.edu.tr

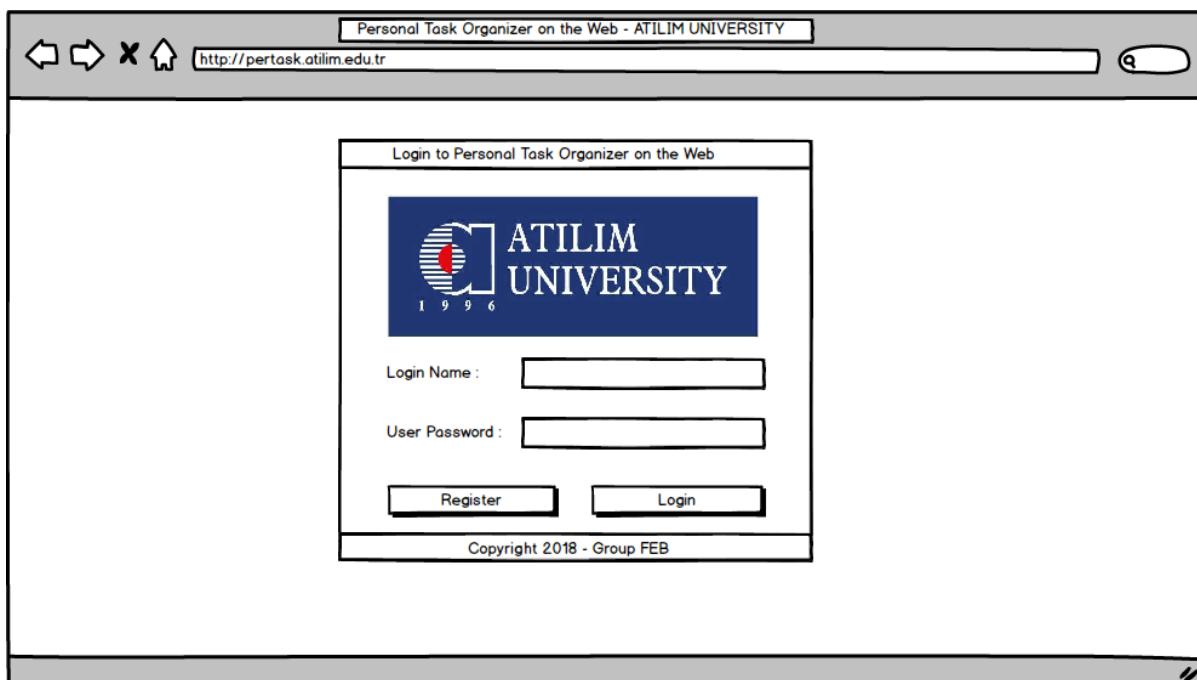
Login to Personal Task Organizer on the Web

ATILIM UNIVERSITY
1996

Login Name :

User Password :

Copyright 2018 - Group FEB



Personal Task Organizer on the Web - ATILIM UNIVERSITY
http://pertask.atilim.edu.tr

Register to Personal Task Organizer on the Web

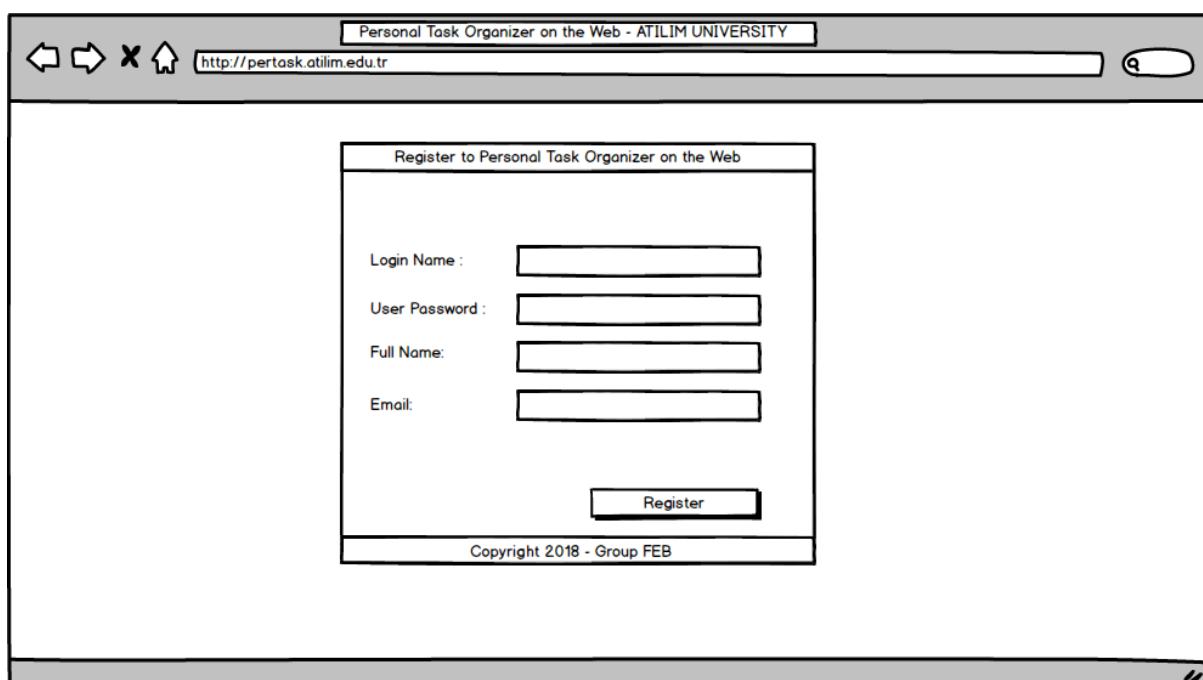
Login Name :

User Password :

Full Name:

Email:

Copyright 2018 - Group FEB





4.4.2 Main Menu- Home Page

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects

- Project-1
- Project-2
- Project-3
- Project-4

Project-1 - Due (15.11.2018) - Active (started)
=====

Project-2 - Due Today (14.11.2018) - On hold (waiting)
=====

Project-3 - Due Tomorrow (16.11.2018) - Planned (not started)
=====

Project-4 - No Deadline (xx.xx.xxxx) - No scheduled
=====

4.4.3 Main Menu- Summary

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects

Deadline	Status	Priority	Owner	UserGroup	Project/Task
15.11.2018	Project	Urgent	Salih Eren TAŞTAN	Students	Project-1
16.11.2018	Planned	Unknown	Furkan YILDIZ	Students	Task-11
17.11.2018	Completed	Low	Salih Eren TAŞTAN	Students	==> Task-11.1
20.11.2018	New	Low	Salih Eren TAŞTAN	Students	==> Task-11.2
18.11.2018	Active	High	Furkan YILDIZ	Students	=> Task-12
xx.xx.xxxx	Project	Unkown	Salih Eren TAŞTAN	Students	Project-2



4.4.4 Main Menu- ToDo List Page

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

ToDo List

ToDo List for Users

- Eren
- Furkan
- Berk

ToDo List for User Groups

- Teachers
- Students
- Personnel

4.4.5 Main Menu- ToDo List Page (ToDo List for the selected User)

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

ToDo List

ToDo List for Users

- Eren
- Furkan
- Berk

ToDo List for User Groups

- Teachers
- Students
- Personnel

ToDo List for the User: Eren

=====

Project-1

=> Task 1.1

=> Task 1.1.1

=> Task 1.1.2

=> Task 1.2

=> Task 1.2.1



4.4.6 Main Menu- ToDo List Page (ToDo List for the selected User Group)

The screenshot shows the 'ToDo List' page. On the left, there is a sidebar with the following menu items:

- Main
 - Home
 - Summary
 - To Do List
 - Calendar
 - Logout
- Projects/Tasks
 - Add project
 - Edit project
 - Add task
 - Edit task
- Task Groups
 - Add
 - Manage
- Users
 - Add
 - Manage
 - Edit my Details
 - Show my Details
- User Groups
 - Add
 - Manage

The main content area is titled 'ToDo List' and contains three sections:

- 'ToDo List for Users' containing the names Eren, Furkan, and Berk.
- 'ToDo List for User Groups' containing the groups Teachers, Students, and Personel.
- 'ToDo List for the User Group: Students' which lists tasks under two projects:
 - Project-1
 - => Task 1.1
 - => Task 1.2
 - => Task 12.1
 - => Task 12.2
 - Project-2
 - => Task 2.1

A red arrow points from the 'To Do List' item in the sidebar to the 'ToDo List for Users' section. Another red arrow points from the 'Students' group in the 'ToDo List for User Groups' section down to the 'ToDo List for the User Group: Students' section.

4.4.7 Main Menu- Add Project Page

The screenshot shows the 'Add a new project' page. On the left, there is a sidebar with the following menu items:

- Main
 - Home
 - Summary
 - To Do List
 - Calendar
 - Logout
- Projects/Tasks
 - Add project
 - Edit project
 - Add task
 - Edit task
- Task Groups
 - Add
 - Manage
- Users
 - Add
 - Manage
 - Edit my Details
 - Show my Details
- User Groups
 - Add
 - Manage

The main content area is titled 'Add a new project' and contains the following fields:

- Project Name:
- Project Creation Date:
- Creation date will be written automatically
- Urgent dropdown: Unknown, Normal, High, Urgent (Urgent is selected)
- Active dropdown: Active, Planned, on Hold, no Deadline (Active is selected)
- Eren dropdown: Eren, Administator (Eren is selected)
- Students dropdown: All Groups, Students, Teachers, Personnel (All Groups is selected)
- All users can see
- User Group members can edit
- Project Description:
- Add Project button

A red arrow points from the 'Add project' item in the sidebar to the 'Add a new project' form. A callout bubble points from the 'Creation date will be written automatically' text to the project creation date field.

**4.4.8 Main Menu- Calendar Page (for the tasks of a selected user in selected month)**

The screenshot shows the main menu of the web application. On the left, there is a sidebar with several sections: Main (Home, Summary, To Do List, **Calendar**, Logout), Projects/Tasks (Add project, Edit project, Add task, Edit task), Task Groups (Add, Manage), Users (Add, Manage, Edit my Details, Show my Details), and User Groups (Add, Manage). A red arrow points from the 'Calendar' link in the Main section to the 'Calendar for Users' section on the right. The 'Calendar for Users' section displays a list of users: Eren, Furkan, and Berk. Below this is a date selector labeled 'November - 2018' with a calendar icon. The main calendar grid shows the days of November 2018. A red arrow points from the date selector to the grid. In the center of the grid, the date '15' is highlighted, and the text 'Task-1.1.1' is displayed in blue. The grid continues with other dates and their corresponding days of the week.

4.4.9 Main Menu- Calendar Page (for the tasks of a selected user group in selected month)

This screenshot is similar to the previous one but shows the calendar for a selected user group. A red arrow points from the 'Calendar' link in the Main section to the 'Calendar for User Groups' section on the right. The 'Calendar for User Groups' section displays a list of groups: Students, Teachers, and Personnel. Below this is the same date selector 'November - 2018'. A red arrow points from the date selector to the grid. In the center of the grid, the date '15' is highlighted, and the text 'Task-1.1.1' is displayed in blue. Another red arrow points from the date '15' to the text 'Task-1.2.1' in the same blue color, indicating a task scheduled for that day. The grid follows the same layout as the previous screenshot.



4.4.10 Main Menu- Summary Page (Select a Project)

The screenshot shows the main menu of the web application. On the left, there's a sidebar with links for Home, Summary, To Do List, Calendar, Logout, Add project, Edit project, Add task, Edit task, Add, Manage, Add my Details, Show my Details, Add, Manage, and Add my Details. The 'Summary' link is highlighted with a red box and an arrow. The main content area is titled 'Projects' and contains a table with columns: Deadline, Status, Priority, Owner, User Group, and Project/Task. The table data is as follows:

Deadline	Status	Priority	Owner	User Group	Project/Task
15.11.2018	Project	Urgent	Salih Eren TAŞTAN	Students	Project-1 Task-1.1 ==> Task-11.1 ==> Task-11.2 => Task-12
16.11.2018	Planned	Unknown	Furkan YILDIZ	Students	
17.11.2018	Completed	Low	Salih Eren TAŞTAN	Students	
20.11.2018	New	Low	Salih Eren TAŞTAN	Students	
18.11.2018	Active	High	Furkan YILDIZ	Students	
xx.xx.xxxx	Project	Unkownn	Salih Eren TAŞTAN	Students	Project-2

Project Details Page:

The screenshot shows the 'Project Details' page for 'Project-3'. The sidebar remains the same as the previous screenshot. The main area has a title 'Project Details:' and a dropdown menu showing 'Project-3' (selected), 'Project-1', 'Project-2', and 'Project-3'. Below the dropdown are fields for 'Project Completion %75' (with a progress bar), 'Project Creation Date: 15/11/2018', 'Project Deadline: 16/11/2018', 'Project Owner: Eren', and 'User Group: Students'. There are two large text input boxes for 'Project Description' and 'Tasks'. At the bottom are 'Edit Project' and 'Add Task' buttons.



4.4.11 Project / Tasks Menu- Edit Project Page (Submit Changes)

Personal Task Organizer on the Web - ATILIM UNIVERSITY <http://>

Main
Home
Summary
To Do List
Calendar
Logout

Projects/Tasks
Add project
Edit project (Red arrow points here)
Add task
Edit task

Task Groups
Add
Manage

Users
Add
Manage
Edit my Details
Show my Details

User Groups
Add
Manage

Edit project

Project Name: Project-3 Parent: None Creation Date: 15/11/2018 (Creation date will be written automatically)

Urgent: Unknown Active Planned on Hold no Deadline

Eren: Administrator Eren Furkan Berk

Students: All Groups Students Teachers Personnel

All users can see
 User Group members can edit (Select one option from pulldown menus)

Project Description: (Enter description of the project here)

Delete Project **Submit Changes** (Red arrow points here)

Project Details Page:

Personal Task Organizer on the Web - ATILIM UNIVERSITY <http://>

Main
Home
Summary
To Do List
Calendar
Logout

Projects/Tasks
Add project
Edit project (Red arrow points here)
Add task
Edit task

Task Groups
Add
Manage

Users
Add
Manage
Edit my Details
Show my Details

User Groups
Add
Manage

Project Details:

Project-3 Project Completion: %75

Project-1 Project Creation Date: 15/11/2018

Project-2 Project Deadline: 16/11/2018

Project-3 Project Owner: Eren

Project-3 User Group: Students

Project Description:

Tasks:

Edit Project **Add Task**



4.4.12 Project / Tasks Menu- Add Task Page

The screenshot shows the 'Personal Task Organizer on the Web - ATILIM UNIVERSITY' interface. On the left, a sidebar menu includes 'Main' (Home, Summary, To Do List, Calendar, Logout), 'Projects/Tasks' (Add project, Edit project, Add task, Edit task, selected), 'Task Groups' (Add, Manage), 'Users' (Add, Manage, Edit my Details, Show my Details), and 'User Groups' (Add, Manage). The 'Add task' button in the sidebar is highlighted with a red box and a red arrow points to it from the bottom right. The main content area is titled 'Add Task' and contains fields for 'Project Name' (Project-3), 'Parent Task' (Task-1.1), 'Task Name' (Task-1.1.1), and 'Creation Date' (15/11/2018). Below these are dropdowns for 'Urgent' (Urgent, Unknown, Normal, High, selected), 'Active' (Active, Planned on Hold no Deadline, selected), 'Eren' (Administrato, Eren, Furkan, Berk, selected), and 'Students' (All Groups, Students, Teachers, Personnel, selected). A tooltip says 'Enter Task Name here' and another says 'Creation date will be written automatically'. At the bottom right of the main area is a large 'Add Task' button, also highlighted with a red box and a red arrow pointing to it from the bottom right.

Project Details Page (with added Task and Subtask):

The screenshot shows the 'Personal Task Organizer on the Web - ATILIM UNIVERSITY' interface. The sidebar is identical to the previous screenshot. The main content area is titled 'Project Details' and displays project information: Project Name (Project-3), Project Completion (%75), Project Creation Date (15/11/2018), Project Deadline (16/11/2018), Project Owner (Eren), and User Group (Students). Below this is a 'Project Description' field and a 'Tasks' field containing 'Task-1.1 ==> Task-1.1.1'. At the bottom right of the main content area is an 'Edit Project' button, and at the very bottom right of the entire page is another 'Edit Project' button.



4.4.13 Main Menu- Summary Page (Select a Task or Subtask)

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary**
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Projects

Deadline	Status	Priority	Owner	User Group	Project/Task
15.11.2018	Project	Urgent	Salih Eren TAŞTAN	Students	Project-1
16.11.2018	Planned	Unknown	Furkan YILDIZ	Students	Task-1.1
17.11.2018	Completed	Low	Salih Eren TAŞTAN	Students	==> Task-1.1.1
20.11.2018	New	Low	Salih Eren TAŞTAN	Students	==> Task-1.1.2
18.11.2018	Active	High	Furkan YILDIZ	Students	=> Task-1.2
xx.xx.xxxx	Project	Unkownn	Salih Eren TAŞTAN	Students	Project-2

Task Details Page (with added Subtask):

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Details:

Task Name:

Project Selection: Project-3

Task Completion: 65%

Task Creation Date: 15/11/2018

Task Deadline: 16/11/2018

Task Owner:

User Group:

Task Description:

Sub Tasks:

- ==> SubTask 1.1.1
- =====> Sub Task 1.1.2.1
- ==> SubTask 1.1.2
- =====> Sub Task 1.1.2.1
- =====> Sub Task 1.1.2.2

Buttons: Edit Task, Add Sub Task



4.4.14 Project / Tasks Menu- Edit Task Page (Submit Changes)

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task**

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Edit Task

Project Name: Project-3 Parent Task: Task-11 Task Name: Task-1.1 Creation Date: 15/11/2018

Urgent Active Eren Students

Unknown Normal Eren Administrato All Groups
High Urgent Planned on Hold Berk Furkan Teachers Personel

Enter Task Name here

Creation date will be written automatically

All users can see
 User Group members can edit

Select one option from pulldown menus

Task Description:

Enter description of the task here

Delete Task Submit Changes

Task Details Page (after edit):

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Task Details

Task Name: Task-1.11 Task Completion: %50

Project-3 Project-1 Project-2 Project-3

Task Creation Date: 15/11/2018

Task Deadline: 16/11/2018

Task Owner: Eren

User Group: Students

Task Description:

Sub Tasks:

==> SubTask 1.1.1
=====> Sub Task 1.1.2.1
==> SubTask 1.1.2
=====> Sub Task 1.1.2.1
=====> Sub Task 1.1.2.2

Edit Task Add Sub Task



4.4.15 Task Groups Menu- Add Task Group Page (Add Task Group)

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add**
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Add Task Group

Task Group (Category) Name: School Tasks

Task Group (Category) Description: All tasks to be done at school

Add Task Group (Category)

4.4.16 Task Groups Menu- Manage Task Group (Edit)

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Manage**

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Manage Task Groups

Name	Description	Action
Home Tasks	All tasks to be done at home	Edit
Work Tasks	All tasks to be done at work	Edit
School Tasks	All tasks to be done at school	Edit

Add new Task Group

**4.4.17 Task Groups Menu- Manage -Edit Task Group Page (Delete)**

The screenshot shows the 'Edit Task Group' page of the Personal Task Organizer on the Web - ATILIM UNIVERSITY. On the left, there is a sidebar with the following menu items:

- Main
 - Home
 - Summary
 - To Do List
 - Calendar
 - Logout
- Projects/Tasks
 - Add project
 - Edit project
 - Add task
 - Edit task
- Task Groups
 - Add
 - Manage** (highlighted with a red box)
- Users
 - Add
 - Manage
 - Edit my Details
 - Show my Details
- User Groups
 - Add
 - Manage

The main content area is titled 'Edit Task Group'. It contains the following fields:

- Task Group (Category) Name: School Tasks
- Task Group (Category) Description: All tasks to be done at school

At the bottom, there are two buttons: 'Submit Changes' and 'Delete Task Group' (highlighted with a red box). A red arrow points from the 'Manage' link in the sidebar to the 'Delete Task Group' button.

4.4.18 Task Groups Menu- Manage Task Group Page (After deleting a Task Group)

The screenshot shows the 'Manage Task Groups' page of the Personal Task Organizer on the Web - ATILIM UNIVERSITY. On the left, there is a sidebar with the same menu items as the previous screenshot.

The main content area is titled 'Manage Task Groups' and displays a table of task groups:

Name	Description	Action
Home Tasks	All tasks to be done at home	Edit
Work Tasks	All tasks to be done at work	Edit

At the bottom of the table, there is a button labeled 'Add new Task Group'.



4.4.19 Users Menu- Add User Page

Personal Task Organizer on the Web - ATILIM UNIVERSITY <http://>

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/ Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Add User

Login Name: eren

Full Name: Salih Eren TAŞTAN

Password: *****

Email: tastan.seren@student.atilim.edu.tr

User Language:

User Type:

User Group:

Select one

Select one

Select one or more

Add User

4.4.20 Users Menu- Manage User Page (After adding a User)

Personal Task Organizer on the Web - ATILIM UNIVERSITY <http://>

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/ Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Manage Users

Name	Action
Administrator	Edit
Salih Eren TAŞTAN	Edit
Furkan YILDIZ	Edit
Berk KELEŞOĞLU	Edit

Add new user



4.4.21 Users Menu- Edit User Page

Personal Task Organizer on the Web - ATILIM UNIVERSITY

http://

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Edit User

Login Name: eren

Full Name: Salih Eren TASTAN

User Password: *****

Email: tastan.seren@student.atilim.edu.tr

User Language:

User Type:

User Group:

Select one

Select one

Select one or more

Submit Changes

Delete User

4.4.22 Users Menu- Manage Users Page (After deleting a User)

Personal Task Organizer on the Web - ATILIM UNIVERSITY

http://

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Manage Users

Name	Action
Administrator	Edit
Furkan YILDIZ	Edit
Berk KELEŞOĞLU	Edit

Add new user



4.4.23 Users Menu- Show my Details Page

The screenshot shows the 'Personal Task Organizer on the Web - ATILIM UNIVERSITY' interface. The left sidebar contains links for Main (Home, Summary, To Do List, Calendar, Logout), Projects/Tasks (Add project, Edit project, Add task, Edit task), Task Groups (Add, Manage), and Users (Add, Manage, **Edit my Details**, **Show my Details**). The main content area has two sections: 'My Details' and 'My Projects/Tasks'. The 'My Details' section displays user information in a table:

Login Name:	eren
Full name:	Salih eren TAŞTAN
Email:	tastan.seren@student.atilim.edu.tr
Admin:	Yes
Normal User:	No
Guest User:	No
User Groups:	Students
Last time seen here:	15.11.2018 - 22:00
Number of projects owned:	3
Number of task owned:	5

The 'My Projects/Tasks' section lists projects and tasks:

Project-1	Project
Task-1.1	Completed on 15.11.2018
Task 1.1	Active (working on it)
Task 1.2	On Hold on 16.11.2018
Project-2	Planned (not active)

4.4.24 Users Menu- Edit my Details Page (User Edit Page)

The screenshot shows the 'Personal Task Organizer on the Web - ATILIM UNIVERSITY' interface. The left sidebar contains links for Main (Home, Summary, To Do List, Calendar, Logout), Projects/Tasks (Add project, Edit project, Add task, Edit task), Task Groups (Add, Manage), and Users (Add, Manage, **Edit my Details**, **Show my Details**). The main content area is titled 'Edit User' and contains a form with the following fields:

Login Name:	eren
Full Name:	Salih Eren TAŞTAN
User Password:	*****
Email:	tastan.seren@student.atilim.edu.tr

Below the form are three dropdown menus with validation messages:

- User Language: Turkish, English (Select one)
- User Type: Admin, Normal, Guest (Select one)
- User Group: Students, Teachers, Personnel (Select one or more)

At the bottom are 'Submit Changes' and 'Delete User' buttons.



4.4.25 Users Group Menu- Add User Group Page

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add**
- Manage

Add User Group

User Group Name: Students

User Group Description: All students at school

User Group Members:

User: **Select one or more**

- Administrator
- Salih Eren TAŞTAN
- Furkan YILDIZ
- Berk KELEŞOĞLU

Add User Group

4.4.26 Users Group Menu- Manage User Groups Page

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add**
- Manage**

Manage User Groups

Name	Description	Action
Students Salih Eren TAŞTAN Furkan YILDIZ Berk KELEŞOĞLU	All students at ATILIM UNIVERSITY	Edit
Teachers Bilge SAY İbrahim CERECİ Cansu Çiğdem EKİN	All teachers working at ATILIM UNIVERSITY	Edit
Personel Ayşegül ÖKE Tariq ÇELİK	All personnel working at ATILIM UNIVERSITY	Edit

Add new User Group



4.4.27 Users Group Menu- Edit User Group Page

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Edit User Group

User Group Name: Students

User Group Description: All students at school

User Group Members:

User:

Administrator
Salih Eren TASTAN
Furkan YILDIZ
Berk KELEŞOĞLU

Submit Changes

Delete User Group

4.4.28 Users Group Menu- Manage User Groups Page (After deleting a User Group)

Personal Task Organizer on the Web - ATILIM UNIVERSITY

Main

- Home
- Summary
- To Do List
- Calendar
- Logout

Projects/Tasks

- Add project
- Edit project
- Add task
- Edit task

Task Groups

- Add
- Manage

Users

- Add
- Manage
- Edit my Details
- Show my Details

User Groups

- Add
- Manage

Manage User Groups

Name	Description	Action
Teachers	All teachers working at ATILIM UNIVERSITY	Edit
Bilge SAY İbrahim CERECİ Cansu Çiğdem EKİN		
Personel	All personnel working at ATILIM UNIVERSITY!	Edit
Ayşegül ÖKE Tarık ÇELİK		

Add new User Group



5. Implementation and Testing

5.1 Implementation

The system has been developed in PHP and HTML. JavaScript, bootstrap, jQuery and Ajax technologies are also utilized. IntelliJ – phpStorm v.2018.2.3 (free for students) has been used as our Integrated Development Environment (IDE). Notepad++ has also been used.

```
// Developer: Group FEB
session_start();
include('menu.php');
include('header.php');
include_once("connect_db.php");

$searchedId = "";
$searchedCategory = "";
$searchedColor = "2111CE";
$searchedTextColor = "#FFEB3B";

$user = isset($_SESSION['user_name']) ? $_SESSION['user_name'] : '';

//add function:
if ( isset($_POST['addcategory'])){

    $category = $_POST['category'];
    $color = "#" . $_POST['color'];
    $textcolor = "#" . $_POST['textcolor'];
    if ( isset($_POST['parent'])) {
        $parentid = $_POST['parent'];
    }else {
        $parentid = NULL;
    }
    $user = $user;

    //Let's remove unwanted HTML (scripting attacks):
    $category = htmlspecialchars($category);

    $sql = "SELECT * FROM categories WHERE user='$user'";
    $query = $connect_db->prepare( $sql );
    $sth = $query->execute();
    $rows = $query->fetchAll( PDO::FETCH_ASSOC );

    foreach($rows as $row) {
        if($row['category'] == $category) {
```



Although we worked locally at home and school, we have shared our codes at **GitHub** environment (Taştan, Yıldız and Keleşoğlu, 2019).

The image displays three separate GitHub repository pages for the group "Group FEB".

- Group FEB**: This repository contains two projects:
 - for-final-report**: Updated 19 hours ago.
 - Atilim-and-FEB-logos**: Updated a day ago. Description: Atılımın logosunun arka planındaki beyaz kısım kaldırıldı.
- pertaskweb_v4.0**: Web'de Kişi Sel Görev Düzenleyici -v4.0 (Türkçe). Updated 5 days ago.
- pertaskweb_v3.0**: PertaskWeb. Updated 12 days ago. Languages: HTML, GPL-3.0.
- pertaskweb_v2.0**: Personal Task Organizer on the Web v.2.0 (User Tutorial is included in the help directory). Updated 12 days ago. Languages: PHP, GPL-3.0.
- homepage**: Updated 16 days ago. Languages: HTML.
- home-page**: Updated 16 days ago. Languages: HTML.

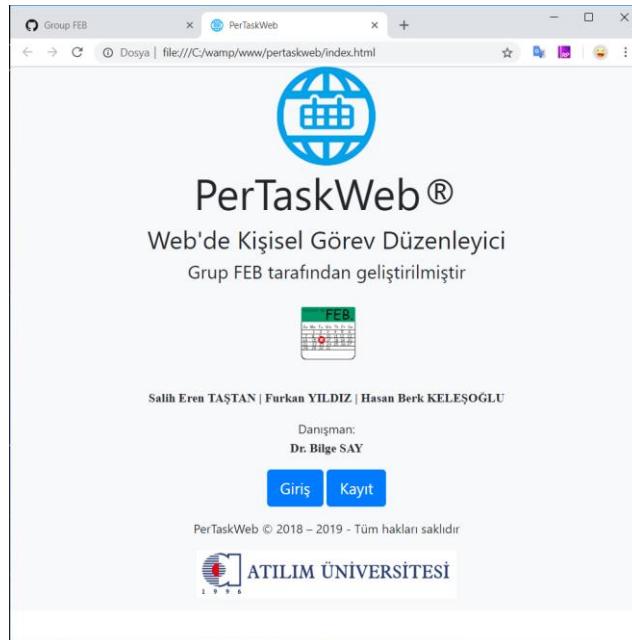
In the sidebar of the first repository, there is a "People" section showing contributors: **erentastan** (Eren TASTAN) and **ghostone06** (Hasan Berk KELEŞOĞLU). There is also a "Top languages" section showing PHP and HTML.

The third repository, **pertaskweb**, shows the following projects:

- pertaskweb**: Personal Task Organizer on the Web (User Tutorial is included in the help directory). Updated 19 days ago. Languages: PHP, GPL-2.0.
- register_verify_login**: User registers, system generates a verification code, writes to database and sent it to the user's email, user verifies (sends) his email and this code, system checks the code in database and the code which user sent, if two codes are the same, the system allows the user to enter the system. Updated 27 days ago. Languages: PHP.
- php_samples**: php samples for the group members. Updated on 26 Nov 2018. Languages: PHP.



The system does not need any installation of a database management system. An empty SQLite database file is in the system. The system can be installed to any PHP supported web server. The only thing to install the system is to copy pertakweb directory to the web serving directory (e.g. c:\wamp\www\pertaskweb) and running the system using a browser (for local use : <http://localhost.pertaskweb>).



The system can also be installed to a remote PHP supported web hosting site (as we have used free hosting site [tr.000webhost.com](https://files.000webhost.com) and uploaded the pertaskweb directory content)

	Name	Size	Date	Permissions
	calendar		2019-01-02 06:57:00	drwxr-xr-x
	database		2019-01-02 11:05:00	drwxr-xr-x
	help		2019-01-02 00:43:00	drwxr-xr-x
	images		2018-12-26 01:06:00	drwxr-xr-x
	js		2019-01-02 06:58:00	drwxr-xr-x
	register		2019-01-02 07:57:00	drwxr-xr-x
	setup		2019-01-02 06:59:00	drwxr-xr-x
	.htaccess	0.3 kB	2019-01-02 08:27:00	-rw-r--r--
	index.html	2.1 kB	2019-01-02 06:59:00	-rw-r--r--



Then the system at remote server can also be used by public at project web site

<http://pertaskwebatilim.000webhostapp.com/> :

The screenshot shows a web browser window with four tabs open: 'Group FEB', 'PerTaskWeb', '000webhost', and 'PerTaskWeb'. The main content area displays the PerTaskWeb logo (a blue globe icon) and the text 'PerTaskWeb ®' followed by 'Web'de Kişisel Görev Düzenleyici'. Below this, it says 'Grup FEB tarafından geliştirilmiştir' and lists three names: Salih Eren TAŞTAN | Furkan YILDIZ | Hasan Berk KELEŞOĞLU. It also credits 'Danişman: Dr. Bilge SAY'. At the bottom are two buttons: 'Giriş' and 'Kayıt'. The footer contains the Atılım University logo with the year '1996', the text 'ATILIM ÜNİVERSİTESİ', and a note 'Powered by 000webhost'.



An on-line responsive HTML help system and also an off-line help file has also been prepared using Free HelpNDoc software (<https://www.helpndoc.com/>). This help system explaines how to install, connect and use the system in detail.

Group FEB | PerTaskWeb | PerTaskWeb | Introduction | +

localhost/pertaskweb/help/Introduction.html

Contents Index Search

0_help_for_pertaskweb

Introduction

Well come to the help page for the Personal Task Organizer on the Web.

About Developers

How to install? How to Install to a local server? How to Install to a remote server?

How to connect? How to connect to the local server? How to connect to the remote server?

How to use How to register? How to organiize task categories How to organize tasks?

How to see and change your profile? How to see your profile? How to change your password?

How to get help? How to get on-line help? How to get information about the system? How to Logout?

PerTaskWeb®
Personal Task Organizer On The Web
Developed by Group FEB
Salih Eren TAŞTAN | Furkan YILDIZ | Hasan Berk KELEŞOĞLU
in supervision of Dr. Bilge SAY
Login Register
PerTaskWeb © 2018 – 2019 - All rights reserved
ATILIM ÜNİVERSİTESİ

Created with the Personal Edition of HelpNDoc: [Full-featured Documentation generator](#)

Copyright © 2018-2019 by Group FEB at Atılım University (Salih Eren TASTAN, Furkan YILDIZ, Hasan KELESOGLU). All rights reserved.

Gizle Geri İleri Ev Yazdır Seçenekler

İçindekiler Dizin Ara Sık Kullanılanlar

Introduction About Developers

How to install? How to Install to a local server? How to Install to a remote server? How to Register to a Free Hosting Site How to upload PERTSAKWEB to the server

How to connect? How to register? How to login? How to get password you forgot? How to organize tasks?

Workflow WorkFlow for Register WorkFlow for Task Management

Well come to the help page for the Personal Task Organizer on the Web.

PerTaskWeb®
Personal Task Organizer On The Web
Developed by Group FEB
Salih Eren TAŞTAN | Furkan YILDIZ | Hasan Berk KELEŞOĞLU
in supervision of Dr. Bilge SAY
Login Register
PerTaskWeb © 2018 – 2019 - All rights reserved
ATILIM ÜNİVERSİTESİ



The screenshot shows the HelpNDoc Personal Edition interface with the following details:

- Project Settings:** Project title: 0_help_, Author: Group FEB, Default topic: Intrc, Version: 5.0, Copyright: Copyright © 2018-2019 by Group FEB a, Summary: [empty], Comment: [empty].
- Language Settings:** Project language: In, Project charset: W, Date / Time format: Customize...
 - Automated settings:** Always synchronize Help ID with topic caption (checked), Synchronize now..., Compress un-compressed included library pictures (checked).
- Library:** A list of library items including: atilm, Current Date/Time, Current Page Number, Date, Date - long, Day, Day - long, Flag_of_NATOsvg, Flag_of_Turkeysvg, Flowchartdiagram1, Flowchartdiagram12, FlowchartRegister, FlowchartRegister1, FlowchartTaskManagement, FlowchartTaskManagement1, image001, image002, image003, image0037, image0038, image040 (selected), image041, image042, image044, image046, image048, image050, image052, image054, image055, image059, image05960, image05961, image05962, image05963.



5.1.1 How to register

- Select the **Kayıt** button at the main page:



- Fill in the form with your information. Do not leave blank any textbox. Enter your username, email, name, surname, password (10 chars minimum), retype your password and click the button **create account (Beni Kaydet)**. If the email not registered before, you have to register first.

The screenshot shows the "Kullanıcı Kayıt Formu" (User Registration Form). The form has several input fields:

- Username: "eren" (highlighted with a yellow background)
- Email: "tastan.seren@student.atilim.edu.tr"
- Name: "Eren"
- Surname: "TAŞTAN"
- Password: "*****" (first field)
- Retype Password: "*****" (second field)

At the bottom right of the form is a button labeled "Beni Kaydet" with a small arrow icon. A cursor arrow is pointing towards this button.



- Check your email and copy your verification code. Enter your email and paste your verification code in the "Kayıt Doğrulama Formu" below. If you enter wrong email or wrong code then you get a warning. (**P.A: user interfaces are responsive and compatible with mobile phones**)

PerTaskWeb®
Web'de Kişisel Görev Düzenleyici
Kayıt Doğrulama Formu

tastan.seren@student.atilim.edu.tr

24367665123

Kaydımı Doğrula

- Select the **Login (Giriş)** option from the main menu:

PerTaskWeb®
Web'de Kişisel Görev
Düzenleyici

Grup FEB tarafından geliştirilmiştir
Salih Eren TAŞTAN | Furkan YILDIZ | Hasan Berk
KELEŞOĞLU

Danişman:
Dr. Bilge SAY

Giriş Kayıt

PerTaskWeb © 2018 – 2019 - Tüm hakları saklıdır

ATILIM ÜNİVERSİTESİ

localhost/pertaskweb/register/login_form.php



- Fill in your username (kullanıcı adı) and password (Şifre) , click the login (Giriş) button:

PerTaskWeb®
Web'de Kişisel Görev
Düzenleyici
Kullanıcı Giriş Formu

eren

.....

Giriş Şifremi Unuttum
Kaydımı Doğrula

Ana Sayfa

- Be sure that you have logged in successfully and click the Continue (Devam) button:

PerTaskWeb®
Web'de Kişisel Görev Düzenleyici

Sisteme başarıyla giriş yaptınız. Devam Etmek için
aşağıdaki butona tıklayınız...

Devam

localhost/pertaskweb/calendar/index.php



- Now you are at the Home page of the **PerTaskWeb** and you see **only your** categories and tasks per category in a **TREE VIEW**:



How to get password you forgot?

- Select the **Giriş** option from the main menu
- Click the **Şifremi Unuttum** button from the **Kullanıcı Giriş Formu** page:

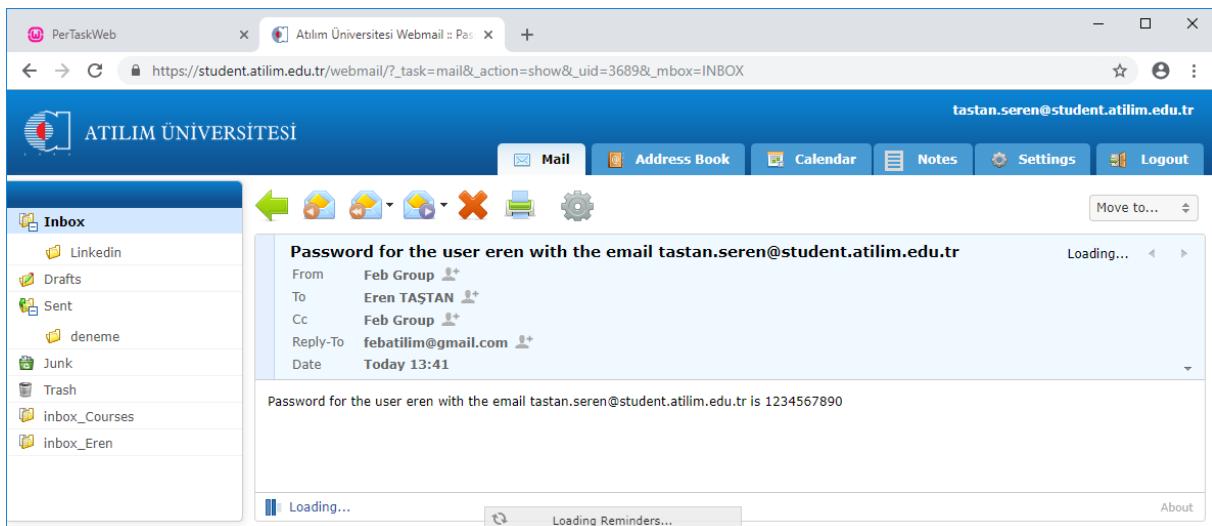
The screenshot shows a web browser window for 'PerTaskWeb'. The title bar says 'Pe'. The main content area has a globe icon, the text 'PerTaskWeb®', 'Web'de Kişisel Görev Düzenleyici', and 'Kullanıcı Giriş Formu'. There are two input fields: one with 'eren' and another with '.....'. Below the fields are three buttons: 'Giriş', 'Şifremi Unuttum' (which is highlighted with a yellow background), and 'Kaydımı Doğrula'. At the bottom is a 'Ana Sayfa' button.



- Enter your email and click the **Send Password (Şifremi Gönder)** button:



- Check your email and get your password:



- Enter your user name and password you get and press **Enter (Giriş)** button.



5.1.2 How to organize task categories

How to add a category?

- Select **My Categories (Kategorilerim) / Add-Edit-Delete Category (Kategori Ekle-Düzenle-Sil)** at the main page :

The screenshot shows a web-based task management application. On the left, there's a sidebar with a tree view of categories: A (expanded), A-1 (expanded), A-1-1 (expanded), A-1-1-b, A-1-1 a, A-1-2, A-2, B (expanded), B-1A, B-1-1, B-1-2, and B-2. On the right, a calendar grid for January 2, 2019, displays various tasks with different colors (red, green, blue, purple, pink). A context menu is open at the top center, with the option 'Kategori Ekle-Düzenle-Sil' highlighted.

- Enter the name of the new category to be added at section **Add Category (Kategori Ekle)**:
If you don't select a **Parent Category (Üst Kategori Seç)** option, then a category at the top level will be created.
You can also select the **background color (Kategori Rengi)** and **text color (Yazı Rengi)** for the new category.
Once you define the parameters for the new category you may enter **Save (Kaydet)** button.



Kategori Ekle

Üst Kategori:

Kategori Adı:

Kategori Rengi: Yazı Rengi:

Kategori Sil

Kategori Seç :

Kategori Düzenle

Kategori Seç:

Kategori Adı:

Kategori Rengi: Yazı Rengi:

- If you want to add a new **child category** (**Alt Kategori**), then you should select the Parent Category from the pulldown menu **Select Parent Category** (**Üst Kategori Seç**):

Kategori Ekle

Üst Kategori:

EXAMS

Kategori Rengi:

Kategori Seç:

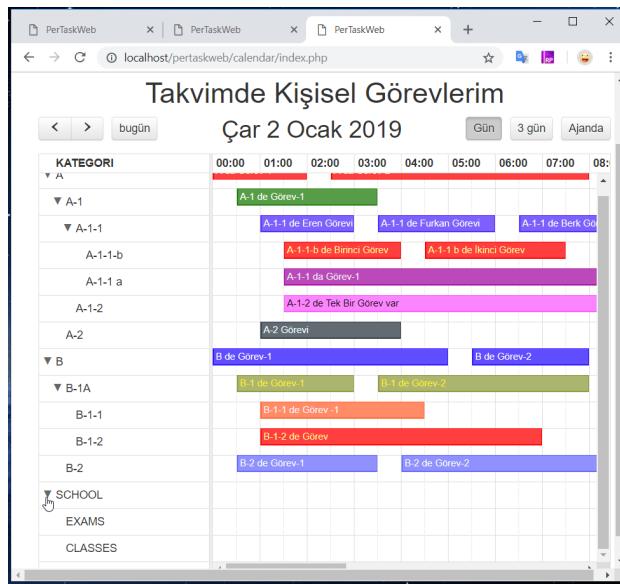
Kategori Seç:

Kategori Adı:

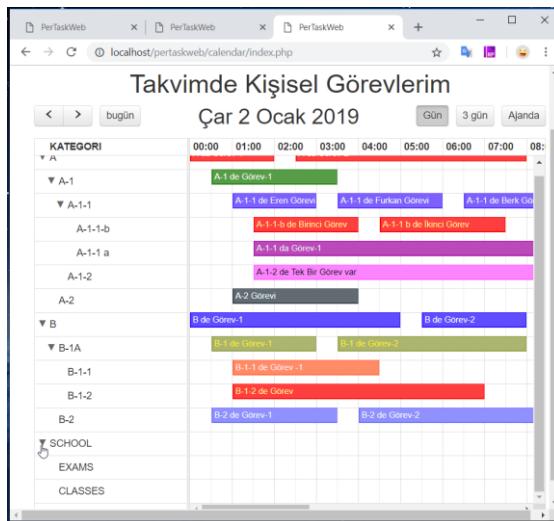
Kategori Rengi: Yazı Rengi:



- Now you can see new parent category (**SCHOOL**) and two child categories (**EXAMS**, **CLASSES**) at the main page:



- Select **My Categories (Kategorilerim)** / **Add-Edit-Delete Category (Kategori Ekle-Düzenle-Sil)** at the main page :



- Select the category from the pulldown menu **Select Category (Kategori Seçin)** at the **Category Delete (Kategori Sil)** section and then click the button **Delete (Sil)**:



Kategori Sil

Kategori Seç : □ SCHOOL

Kategori Düzenle

Kategori Seç : □ SCHOOL

Kategori Adı:

Kategori Rengi: 2111CE Yazı Rengi: FFFEBO

- You can verify that the category (together with its children) and all tasks in this parent and child categories are deleted:

Kategori Sil

Kategori Seç : Kategori Seçin

Kategori Seçin

- A
- B
- A-1
- A-2
- B-1
- B-2
- A-1-1
- A-1-2
- A-1-1-b
- B-1-1
- B-1-2
- A-1-1 a

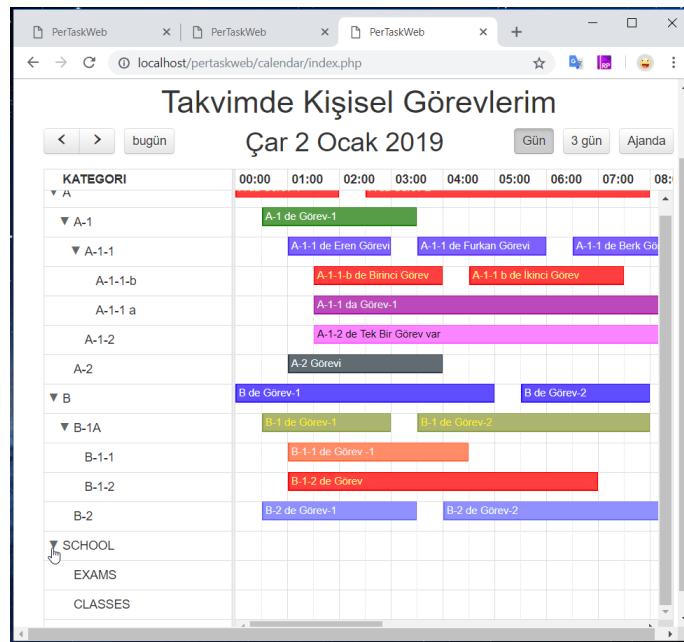
Kategori Adı:

Kategori Rengi:



How to edit a category?

- Select My Categories (Kategorilerim) / Add-Edit-Delete Category (Category Ekle-Düzenle-Sil) at the main page :



- Select the category from the pulldown menu Select Category (Kategori Seçin) at the Category Edit (Kategori Düzenle) section and then click the button Searh (Ara):

- You can notice that the parameters (name, color and text color) of the category to be edited are displayed in the text boxes:



Kategori Ekle

Üst Kategori: Üst kategori seç ▾

Kategori Adı:

Kategori Rengi: FF0000 Yazı Rengi: FFFF69A

Kategori Sil

Kategori Seç : Kategori Seçin ▾

Kategori Düzenle

Kategori Seç: Kategori Seçin ▾

Kategori Adı:

B-1

Kategori Rengi: EAFF68 Yazı Rengi: 000000

- You can change the category parameters and click **Update (Güncelle)** button:

Kategori Ekle

Üst Kategori: Üst kategori seç ▾

Kategori Adı:

Kategori Rengi: FF0000 Yazı Rengi: FFFF69A

Kategori Sil

Kategori Seç : Kategori Seçin ▾

Kategori Düzenle

Kategori Seç: Kategori Seçin ▾

Kategori Adı:

B-1A

Kategori Rengi: #009D40 Yazı Rengi: #FFEE1D

P.A: When you changed the color and text color of category all the colors and text colors of this category are changed automatically.



How to see all categories?

- Select My Categories (Kategorilerim) / All Categories (Tüm Kategoriler):

The screenshot shows a web-based calendar application. At the top, there are three tabs: 'PerTaskWeb' (selected), 'Ana Sayfa', and 'Görevlerim'. Below the tabs, a navigation bar includes 'Kategorilerim', 'Yardım', 'Profilim', and 'Çıkış'. A dropdown menu 'Tüm Kategoriler' is open over the 'Kategorilerim' button. The main area displays a calendar for 'Car 2 Ocak 2019'. On the left, a tree view lists categories: 'A', 'A-1', 'A-1-1', 'A-1-1-b', 'A-1-1 a', 'A-1-2', 'A-2', 'B', 'B-1A', 'B-1-1', 'B-1-2', and 'B-2'. To the right, a grid shows tasks for each category, color-coded by category. For example, category 'A' is red, 'B' is blue, and 'A-1' is green. Task names like 'A da Görev-1', 'A da Görev-2', etc., are visible within their respective colored boxes.

- You will see all your categories in their colors and textcolors:

The screenshot shows a page titled 'Tüm Kategorilerim' (All Categories). At the top, there are tabs: 'PerTaskWeb' (selected), 'Ana Sayfa', 'Görevlerim', 'Kategorilerim' (selected), 'Yardım', 'Profilim', and 'Çıkış'. Below the tabs, a table titled 'Kategori Adı ve Rengi' (Category Name and Color) lists categories and their assigned colors. The categories and colors are: A (red), B (blue), A-1 (green), A-2 (dark green), B-1A (yellow-green), B-2 (purple), A-1-1 (pink), A-1-2 (light pink), A-1-1-b (orange), B-1-1 (orange), B-1-2 (light orange), A-1-1 a (purple), SCHOOL (black), EXAMS (green), CLASSES (black), HOME (red), PROJECT (blue), COMPUTER GAME (dark blue), and EATING (black). A 'Çık' (Exit) button is at the bottom left.

Kategori Adı ve Rengi
A
B
A-1
A-2
B-1A
B-2
A-1-1
A-1-2
A-1-1-b
B-1-1
B-1-2
A-1-1 a
SCHOOL
EXAMS
CLASSES
HOME
PROJECT
COMPUTER GAME
EATING

**How to add a new task?**

- Select the day from the calendar using < or > buttons:

The screenshot shows a web browser window with three tabs, all titled "PerTaskWeb". The central tab displays a calendar titled "Takvimde Kişisel Görevlerim" for "Paz 6 Ocak 2019". The sidebar on the left lists categories: A, B, and SCHOOL, with SCHOOL expanded to show EXAMS and CLASSES. The main area is a grid representing time slots from 10:00 to 18:00. Buttons at the top right include "bugün", "Gün", "3 gün", and "Ajanda".

This screenshot is similar to the one above, but the "SCHOOL" category under the sidebar is now selected, indicated by a mouse cursor icon over its name.

- Select the time period (for example from 14:00 to 17:00) for the sub category (for example EXAMS)



The screenshot shows a web-based calendar application. At the top, there are three tabs labeled 'PerTaskWeb', 'Ana Sayfa' (Home), and 'Görevlerim' (Tasks). Below the tabs, the title 'Takvimde Kişisel Görevlerim' (Personal Tasks in the Calendar) is displayed. The date 'Paz 6 Ocak 2019' is centered above a grid. The grid has columns for time from 13:00 to 20:00 and rows for categories: 'KATEGORİ', '► A', '► B', '▼ SCHOOL', 'EXAMS', and 'CLASSES'. The 'EXAMS' category is expanded, showing 'EXAMS' in the first two columns of the 13:00-14:00 slot.

- Enter the Task (**Görev**) and Category (**Kategori**) (if you wish different category from that you just selected):

The screenshot shows a modal dialog box titled 'Görev Ekle' (Add Task). It contains four input fields: 'Görev' (Task) with the value 'MATH-276 EXAM', 'Kategori' (Category) with a dropdown menu showing 'Başka Kategori Seçin' (Select Other Category), 'Başlangıç' (Start Date/Time) with the value '2019-01-06 14:00:00', and 'Bitiş' (End Date/Time) with the value '2019-01-06 17:00:00'. At the bottom right of the dialog are two buttons: 'Kapat' (Close) and 'Kaydet' (Save), with 'Kaydet' being highlighted by a cursor.



- You can see your new task on the calendar:

The screenshot shows a web-based calendar application. At the top, there are three browser tabs all labeled "PerTaskWeb". The main content area is titled "Takvimde Kişisel Görevlerim" (Personal Tasks in the Calendar) and shows the date "Paz 6 Ocak 2019" (Saturday, January 6, 2019). Below the date are buttons for navigating between days ("< >") and viewing options ("bugün", "Gün", "3 gün", "Ajanda"). A sidebar on the left lists categories: "KATEGORİ" (Category), "► A", "► B", "▼ SCHOOL", "EXAMS", and "CLASSES". The main area is a grid representing time from 13:00 to 20:00. A blue rectangular box represents a task for "EXAMS" in the "SCHOOL" category, titled "MATH-276 EXAM", scheduled for 18:00.

How to change the time period of a task?

- If you wish to change the time period of the task you can select, drag and drop the task along the time line of the category:

This screenshot is identical to the one above it, showing the same calendar interface and the task "MATH-276 EXAM" scheduled for 18:00 on Saturday, January 6, 2019. The purpose of this second screenshot is likely to show the state of the task before or after a potential dragging operation.



How to change the start and end time of a task?

- If you wish to change the start or end time of the task you can select on either side of task, and then drag and drop the side along the time line until the time you wish:

The screenshot shows a web-based calendar application titled "Takvimde Kişisel Görevlerim" (Personal Tasks in the Calendar) for Friday, January 6, 2019. The interface includes navigation buttons for date selection, a sidebar with categories like A, B, and SCHOOL, and a main grid for tasks. A specific task, "MATH-276 EXAM", is highlighted with a blue background and a double-headed arrow indicating it is being moved. The timeline at the bottom shows hours from 13:00 to 20:00.

How to change the category of the task?

- If you wish to change the category of task, you can select the task, drag and drop into the time line of the category you wish:

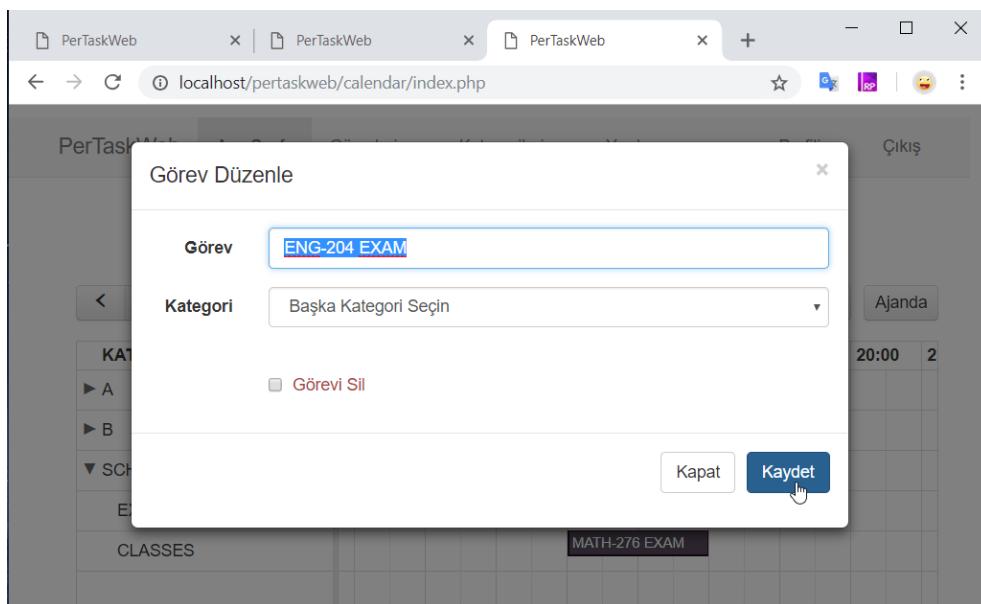
(P.A: You may notice the color and text color of the task are changed according to the new category automatically):

This screenshot shows the same application after the task has been moved. The "MATH-276 EXAM" task is now located in the "EXAMS" category section of the timeline, indicating it has been successfully moved. The task's color and text color have changed to match the new category.



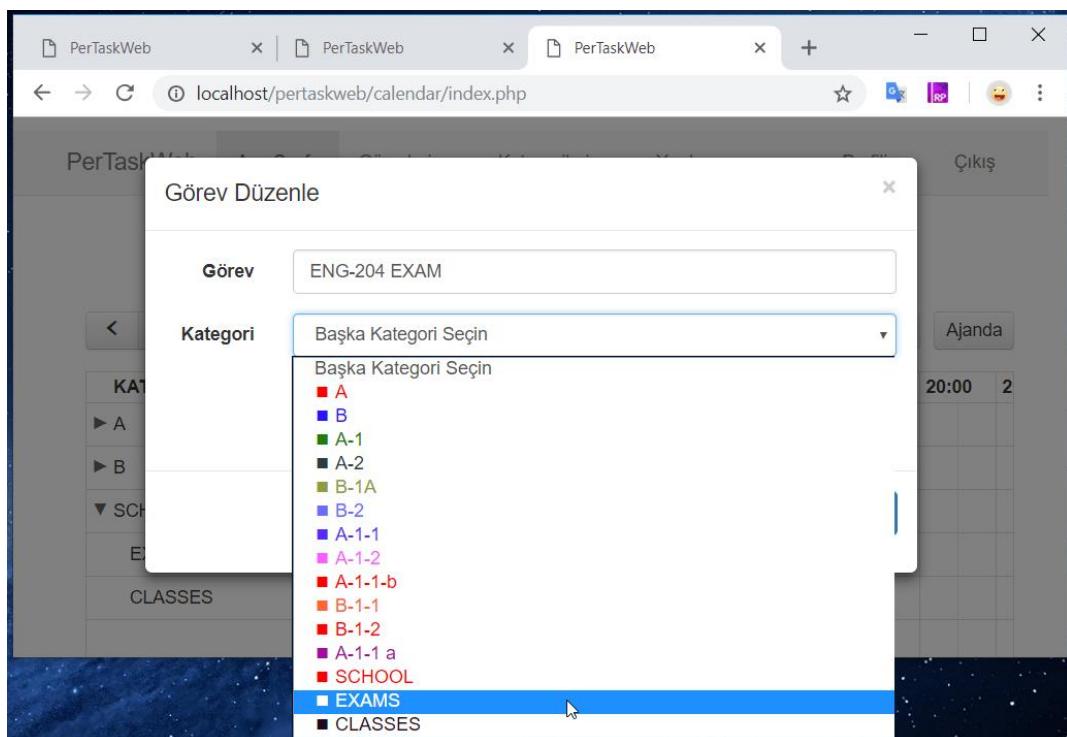
How to change the name of the task ?

- If you wish to change the name of task you may double click on the task and change the name and **Save (Kaydet)**:



How to change the name and category the task together?

- You may double click on the task, change the name and select a different category (**Başka Kategori Seçin**) from the pull-down menu and **save (Kaydet)**:





- You will see the task with the new name and in the timeline of the new category:

The screenshot shows a web-based calendar interface for 'PerTaskWeb'. The title bar says 'Takvimde Kişisel Görevlerim' (Personal Tasks in the Calendar) and 'Paz 6 Ocak 2019' (Sat 6 Jan 2019). The sidebar on the left lists categories: 'KATEGORİ' (Category) with 'A', 'B', and 'SCHOOL' expanded to show 'EXAMS' and 'CLASSES'. The main area is a grid representing time from 14:00 to 21:00. A blue box highlights the 'EXAMS' category in the grid, and a cursor points to the task 'ENG-204 EXAM' at 17:00.

How to change the color and text color of the task ?

- If you wish to change the color and text color of the task, then you should [change the color and text color of the category](#) which the task belongs to:

The screenshot shows a 'Category Edit' page. It has fields for 'Kategori Seç' (Select Category), 'Kategori Adı' (Category Name - 'EXAMS'), and color selection boxes for 'Kategori Rengi' (Category Color - green) and 'Yazı Rengi' (Text Color - yellow). A 'Güncelle' (Update) button is visible, and a cursor is over it.

- You can see the task in the new colors of the category which it belongs to:



The screenshot shows a web-based calendar application. At the top, there are three browser tabs all labeled "PerTaskWeb". The main content area has a header "Takvimde Kişisel Görevlerim" (Personal Tasks in the Calendar) and a date "Paz 6 Ocak 2019". Below the date are buttons for "bugün" (Today), "Gün" (Day), "3 gün" (3 days), and "Ajanda" (Agenda). A sidebar on the left lists categories: "KATEGORİ" (Category) with "► A", "► B", and "▼ SCHOOL" expanded to show "EXAMS" and "CLASSES". The main calendar grid shows a green box for the task "ENG-204 EXAM" at 18:00 on January 6, 2019.

How to delete a task?

- Double click the task to be deleted and check on **Delete Task (Görevi Sil)** option and **Save (Kaydet)**:

The screenshot shows a modal dialog box titled "Görev Düzenle" (Task Edit). It contains fields for "Görev" (Task) set to "ENG-204 EXAM" and "Kategori" (Category) set to "Başka Kategori Seçin" (Select Another Category). There is a checked checkbox labeled "Görevi Sil" (Delete Task). At the bottom right are "Kapat" (Close) and "Kaydet" (Save) buttons. The background shows the same calendar interface as the previous screenshot.

- You will see that the task you deleted has been disappeared on the calendar:



The screenshot shows a web-based calendar application. At the top, there are three tabs labeled 'PerTaskWeb'. Below the tabs, the URL is 'localhost/pertaskweb/calendar/index.php'. The main header reads 'Takvimde Kişisel Görevlerim' (Personal Tasks in the Calendar) and 'Paz 6 Ocak 2019' (Saturday, January 6, 2019). A navigation bar includes buttons for '< >' (month navigation), 'bugün' (Today), 'Gün' (Day), '3 gün' (3 days), and 'Ajanda' (Agenda). On the left, a sidebar lists categories: 'KATEGORİ' (Category) with '► A', '► B', '▼ SCHOOL' expanded to show 'EXAMS' and 'CLASSES', and '▼ HOME'. The main area is a grid representing the day from 13:00 to 20:00. There are no visible task entries.

5.1.3 How to organize tasks

How to see your today's tasks?

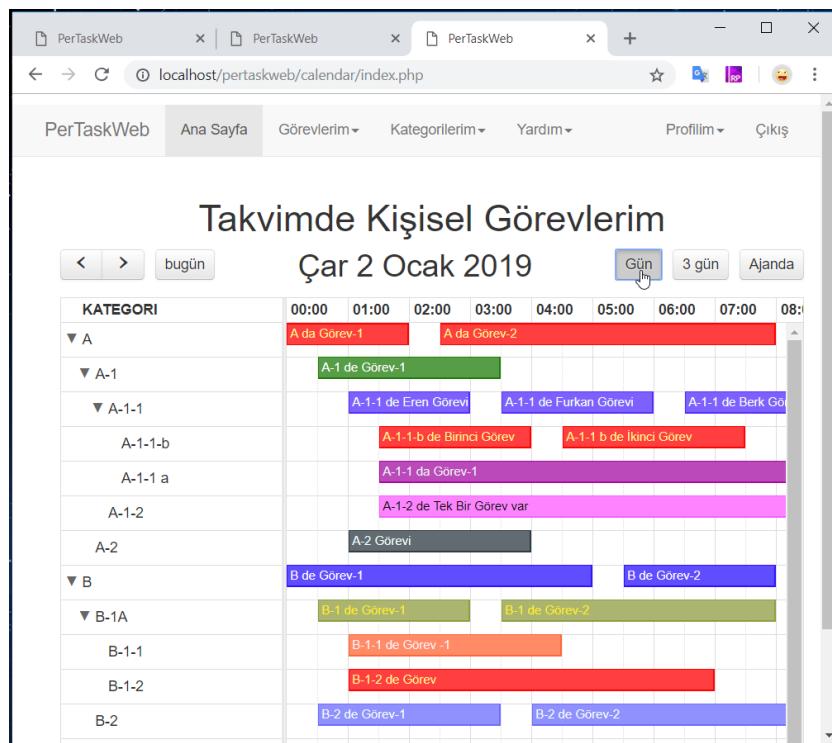
- Select the Today (bugün) option at the upper left:

This screenshot shows the same calendar interface as the previous one, but with the 'bugün' (Today) button highlighted. The main grid now displays specific tasks for the day: 'PROJECT' from 10:00 to 12:00 with the title 'Prepare up-to-date help', 'COMPUTER GAME' from 13:00 to 14:00 with the title 'World of Warcraft', 'EATING' from 14:00 to 15:00 with two entries: 'Breakfast' and 'Lunch'. The other categories and time slots remain empty.



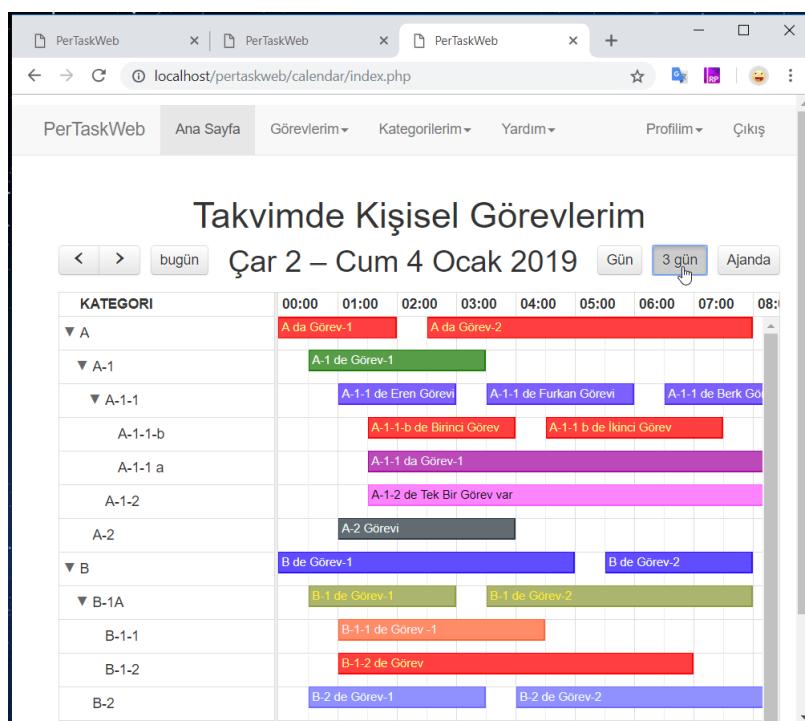
How to see your tasks in day view?

- Select the Day (Gün) option at the upper right:



How to see your tasks in 3 day view?

- Select the 3 Day (3 Gün) option at the upper right:





How to see your tasks in Agenda?

- Select the **Agenda (Ajanda)** option at the upper right:

The screenshot shows a web browser window for 'PerTaskWeb' displaying a calendar for 'Ocak 2019'. The title 'Takvimde Kişisel Görevlerim' is at the top. Below it, there are buttons for navigating between days ('< >'), selecting the current day ('bugün'), viewing 'Gün' (day), '3 gün' (3 days), and 'Ajanda' (agenda). The agenda view lists tasks from 00:00 to 08:30. Each task is preceded by a colored circle representing its category: red, blue, green, purple, black, pink, and dark green. The tasks are: A da Görev-1, A-1 Görevi, A-1 de Görev-1, B-2 de Görev-1, B-1 de Görev-1, B-1-2 de Görev, B-1-1 de Görev -1, A-2 Görevi, A-1-1 de Eren Görevi, A-1-1 da Görev-1, A-1-2 de Tek Bir Görev var, A-1-1-b de Birinci Görev, A da Görev-2, B-1 de Görev-2, A-1-1 de Furkan Görevi, and B-2 de Görev-2.

How to see your tasks on category basis?

- From the main menu, select **My Tasks (Görevlerim) / May Tasks on Category Basis (Kategorilere Göre Görevlerim)**:



The screenshot shows the 'Görevlerim' (Tasks) section of the application. A context menu is open, with the option 'Kategorilere Göre Görevlerim' (Tasks by Category) highlighted. The sidebar on the left lists categories A and B with their respective sub-tasks. The main area displays a grid for tasks from 00:00 to 08:00 on January 19, 2019.

- Select the tab of the category (for example **PROJECT**) you wish to see its tasks and you will see task information:

The screenshot shows the 'Kategorilerime göre Görevlerim' (Tasks by Category) section. The 'PROJECT' category is selected in the navigation bar. Below, a table lists tasks under the PROJECT category, along with their start and end dates.

TASK	START-DATE	END-DATE
Prepare up-to-date help	2019-01-06 10:00:00	2019-01-06 13:00:00
Prepare Final Report	2019-01-06 16:00:00	2019-01-06 23:00:00

Enter the **Exit (Çık)** button to go back to main page.



5.2 Testing

As the group of developers, we tested the system against the functional requirements.

5.2.1 Test Plan

NA

5.2.2 Test Procedures

NA



6. Evaluation

6.1 Problems

- User can not add anything from smartphones (using smartphones for only monitoring)
- If you use on PC you can not use on smart phone at the same time.

6.2 Performance

- Adding tasks and categories are quite fast.
- Sending verification code and forgotten password to the user's email is very efficient.
- On-line help is very useful.

6.3 Usability

- The user interface is easy to use.
- If you have difficulties we have also on-line help is very useful..

6.4 Deployment

- In order to deploy the system, just copy the system directory to the web server (e.g. to c:\wamp\www) or upload it to a PHP supported web host.
- Users should have a personal computer.
- The internet connection is necessary and should be provided.

6.5 Back up & Recovery

- Users are responsible to take their own back ups and apply recovery

6.6 Security

- Unique username and emails are stored in a database, therefore users can not use multiple accounts.
- Users have to verify their account by a verification code which is sent to the system to their emails.
- If users forgot their password they can request their password and the system send their password to their emails.
- Users can change their password at any time.



6.7 Compliance with the Realistic Constraints

The constraints of the system, such as requirement for high performance and fast loading times are met when the system is evaluated. Table 6.1 shows the summary of those constraints and conditions given.

Table 6.1 Compliance with the Realistic Constraints

Economic Factors	NA
EXPENDITURES	
Computer	We used our own personal computers to develop our project
Other Devices	NA
Peripherals	NA
Internet Connection	Currently owned connection was used
Software	For Coding: InteliJ- phpStorm (Free Student Edition) For local host: WAMP server For help preparing: HelpNDoc
Human Resources	Our project team consists of 3 people.
Online Resources	https://www.w3schools.com/ , https://www.ajax-tutor.com/ , http://www.tutorialspoint.com/ , https://www.opensourcevaristy.com , https://getbootstrap.com are used as manuals for php,js,css, html etc. https://github.com/ and https://codereview.stackexchange.com/ for samples. https://www.draw.io/ is used to draw UML and ER diagrams.
FUNDING RESOURCES	
University Resources	NA
Project support (TUBITAK etc)	NA
Support by the Undustry	NA
Self-funded	NA
OTHER CONSTRAINTS	
Time	Each team members' free times do not fit each other.
Knowledge	Each team member have different knowledge and it was now enough to develop with equal efforts.
MANAGERIAL	
Schedule (time)	The software was expected to be completed in nearly 12 weeks but we needed more time to complete project depending on constraints.



7. Conclusion

Our program was planned to developed with the synchronization of the Tony Buzan's method that called mind map. But after we started planing stage we reqlized that it was difficult to achieve for us right now because of the graphical reprentation of the parent and child nodes. So we decided to do treeview lists of task categories in our project by getting some inspiration from mind mapping concept. Our program is developed in PHP using the IDE Intelij-phpStorm (Free Student Edition) and the Web Server WAMP. So we could display our PHP applications easily at local host without any problem. We shared our codes at GitHub environment creating a group called as FEB which is the name of our project group. When the working partial code, as compliant with agile software development method, we uploaded the system to the free PHP supported web hosting portal tr.000webhost.com after registering to that portal first.

As result, we learned a lot with this project and we suggest the software developers use PHP and JavaScript togetherwith the usefull plugins such as JQuery, Bootstrap and Fullcalendar. As for the those who are interested in using this system, they can register and login at the URL of our system at remote host <http://pertaskwebatlim.000webhostapp.com/> and use used free of charge. The users who want to install their own system, can take this system directory (in zipped format) from the GitHub (<https://github.com/GroupFEB>) , unzip the rar file, copy to their local web server or upload to a PHP enabled internet or intranet web hosting portal and let the users to use it. (Taştan, Yıldız and Keleşoğlu, 2019).



References

Brandal,B. (2018) “Get More Done: The Complete Introduction to Task Management”,
<https://www.process.st/wp-content/uploads/2016/11/Get-More-Done-%E2%80%94-The-Complete-Guide-to-Task-Management.pdf>, Accessed on 15/10/2018.

Belotti, V. and others (2004), "What a To-Do: Studies of Task Management Towards the Design of a Personal Task List Manager", Conference Paper, 2004,
https://www.researchgate.net/publication/221518959_What_a_to-do_studies_of_task_management_towards_the_design_of_a_personal_task_list_manager, Accessed on 15/10/2018.

Botla, P. (2013), "Designing Personal Assistant Software for Task Management using Semantic Web Technologies and Knowledge Databases", Working Paper, 2013,
<http://oaji.net/articles/2015/1948-1430994326.pdf>, Accessed on 15/10/2018.

Chasanidou,D and others (2013), “Enabling team collaboration with task management tools”,
<http://www.opensym.org/os2016/proceedings-files/p601-chasanidou.pdf>

Akçalı, A.K (2018), “Interview on Mind Mapping”, HAVELSAN.

Taştan, S.E., Yıldız, F, Keleşoğlu, H.B. (2019), “GroupFEB Organization at GitHub”,
<https://github.com/GroupFEB>, Accessed on 15/01/2019.



Appendices:

Appendix-A: Interview with Dr. Akin Kağan AKÇALI at HAVELSAN

Question-1: What is mind mapping?

Answer-1: Mind mapping is a highly effective way of getting information in and out of your brain. Mind mapping is a creative and logical means of note-taking and note-making that literally "maps out" your ideas.

Question-1: What are the main characteristics of mind maps?

Answer-2: All Mind Maps have some things in common. They have a natural organizational structure that radiates from the center and use lines, symbols, words, color and images according to simple, brain-friendly concepts. Mind mapping converts a long list of monotonous information into a colorful, memorable and highly organized diagram that works in line with your brain's natural way of doing things.

Question-3: Can you give us an example for mind mapping?

Answer-3: One simple way to understand a Mind Map is by comparing it to a map of a city. The city center represents the main idea (root node); the main roads leading from the center represent the key thoughts in your thinking process (child nodes); the secondary roads or branches represent your secondary thoughts, and so on. Special images or shapes can represent landmarks of interest or particularly relevant ideas.

Question-4: How can we use mind mapping for personal task organizing?

Answer-4: Let me show you few examples using my self-created mind mapping tool **Drop Task**. As you see we may draw on a paper our weekly to-do list or we take down lecture notes in a mind map style. Then we can draw these lists or notes by using a mind mapping tool. There is always one root node and many child nodes connected to the root. Each child node may be the parent of other child nodes. You may change the parent of child nodes. You may collapse or expand all child nodes of a parent node. In that way, you may define tasks as parent and subtasks as child nodes.



Appendix-B: Sample Codes of the Systems

B.1 Login Codes

B.1.1 login_form.php

```
<?php
    // Developer: Group FEB
    session_start();
    session_destroy();
    $conn = null;
    $username = "";
    $password = "";
    include('header.php');
    include_once("db_connect.php");
?>
<title>PerTaskWeb</title>
<link href="css/bootstrap.min.css" rel="stylesheet">
<link href="css/product.css" rel="stylesheet">
<script type="text/javascript" src="script/validation.min.js"></script>
<script type="text/javascript" src="script/login.js"></script>
<link href="css/style.css" rel="stylesheet" type="text/css" media="screen">
<div class="blue position-relative overflow-hidden p-0 p-md-0 m-md-0 text-center bg-light">
    <div class="container">
        
        <h1 class="display-4 font-weight-normal">PerTaskWeb®</h1>
        <h3 class="display-5 font-weight-normal">Web'de Kişisel Görev Düzenleyici</h3>
        <div class="register_container">
            <form class="form-signin" method="post" id="login-form">
                <h2 class="form-signin-heading">Kullanıcı Giriş Formu</h2>
                <hr />
                <div id="error">
                </div>

                <div class="form-group">
                    <input type="text" class="form-control" placeholder="Kullanıcı Adınız" name="user_name" id="user_name" />
                </div>
                <div class="form-group">
                    <input type="password" class="form-control" placeholder="$ifreniz" name="password" id="password" />
                </div>
                <hr />
                <div class="form-group">
                    <button type="submit" class="btn btn-default" name="btn-save" id="btn-submit" value="Clicked">
                        <span class="glyphicon glyphicon-log-in"></span> Giriş
                    </button>
                    <button type="button" class="btn btn-default" name="btn-forgot" id="btn-forgot" value="Clicked" onClick="location.href='forgot_form.php'">
                        <span class="glyphicon glyphicon-log-in"></span> Şifremi Unuttum
                    </button>
                    <button type="button" class="btn btn-default" name="btn-verify" id="btn-verify" value="Clicked" onClick="location.href='verify_form.php'">
                        <span class="glyphicon glyphicon-log-in"></span> Kaydımı Doğrula
                    </button>
                </div>
            </form>
            <div style="margin:50px 0px 0px 0px;">
                <a class="btn btn-default read-more" style="background:#3399ff;color:white" href=".//index.html" title="">Ana Sayfa</a>
            </div>
        </div>
    </div>
<?php include ('footer.php');?>
```



B.1.2 login.js

```
// Developer: Group FEB
$(document).ready(function() {
    /* handle form validation */
    $("#login-form").validate({
        rules:
        {
            user_name: {
                required: true,
                minlength: 3
            },
            password: {
                required: true,
                minlength: 8,
                maxlength: 15
            },
        },
        messages:
        {
            user_name: "Lütfen en az 3 karakterli kullanıcı adı yazınız!",
            password:{
                required: "Lütfen şifrenizi yazınız!",
                minlength: "$ifreniz en az 8 ve en fazla 15 karakter olmalı!"
            },
        },
        submitHandler: submitForm
    });

    /* handle form submit */
    function submitForm() {
        var data = $("#login-form").serialize();

        $.ajax({
            type : 'POST',
            url : 'login.php',
            data : data,
            beforeSend: function() {
                $("#error").fadeOut();
                $("#btn-submit").html('<span class="glyphicon glyphicon-transfer"></span>  ');
Giriş başladı...');
            },

            success : function(response) {
                //window.alert(response);
                if(response==1){
                    $("#error").fadeIn(1000, function(){
                        $("#error").html('<div class="alert alert-danger"> <span class="glyphicon glyphicon-info-sign"></span>   Özür dileriz. Kullanıcı adınız veya eposta adresiniz hatalı veya kaydınızı henüz doğrulamadınız!</div>');
                        $("#btn-submit").html('<span class="glyphicon glyphicon-log-in"></span>   Giriş');
                    });
                } else if(response=="logged_in"){
                    $("#btn-submit").html('   Giriş yapılmıyor...');

                    setTimeout('.form-signin').fadeOut(500, function(){
$.register_container.load("welcome.php"); }, 3000);
                } else {
                    $("#error").fadeIn(1000, function(){
                        $("#error").html('<div class="alert alert-danger"><span class="glyphicon glyphicon-info-sign"></span>   '+data+' !</div>');
                        $("#btn-submit").html('<span class="glyphicon glyphicon-log-in"></span>   Giriş');
                    });
                }
            });
        return false;
    }
});
```



B.1.3 login.php

```
<?php
// Developer: Group FEB
session_start();
session_destroy();
include_once("db_connect.php");

if(isset($_POST['btn-save'])) {
    $user_name = $_POST['user_name'];
    $user_password = $_POST['password'];

    //Let's remove unwanted HTML (scripting attacks):
    $user_name = htmlspecialchars($user_name);
    $user_password = htmlspecialchars($user_password);

    $sql = "SELECT * FROM users WHERE user ='$user_name'";

    //PDO (PHP Data Object):
    $query = $conn->query($sql);
    $row = $query->fetch(PDO::FETCH_ASSOC);

    if ((($row['user']) AND ($row['pass'] == $user_password) AND
    ($row['verified'] == "1")) ) {
        session_start();
        $_SESSION['user_name'] = $user_name;
        $_SESSION['logged_in'] = "yes";
        $conn = null;
        echo "logged_in";
    } else {
        echo "1";
    }
}
?>
```



B.2 Main Codes

B.2.1 menu.php (main menu)

```
<!-- Developer: Group FEB -->
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="icon" href="images/pertaskweb_icon.png">
<link href="css/bootstrap.min.css" rel="stylesheet" type="text/css">
</head>
<body class="" style="padding-bottom: 20px; padding-top: 10px;">
    <div role="navigation" class="navbar navbar-default navbar-static-top">
        <div class="container">

            <!-- ===== -->
            <!-- Brand and toggle get grouped for better mobile display -->
            <div class="navbar-header">
                <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-
target="#defaultNavbar1" aria-expanded="false"><span class="sr-only">Toggle
navigation</span><span class="icon-bar"></span><span class="icon-bar"></span><span
class="icon-bar"></span></button>
                <a class="navbar-brand" href="#">PerTaskWeb</a>
            </div>
            <!-- Collect the nav links, forms, and other content for toggling -->
            <div class="collapse navbar-collapse" id="defaultNavbar1">
                <ul class="nav navbar-nav">
                    <li class="active"><a href="index.php">Ana Sayfa<span class="sr-
only">(current)</span></a></li>

                    <li class="dropdown"><a href="#" class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="false">Görevlerim<span
class="caret"></span></a>
                        <ul class="dropdown-menu">
                            <li><a href="index.php">Takvimde Kişisel Görevlerim</a></li>
                            <li role="separator" class="divider"></li>
                            <li><a href="category-tasks.php">Kategorilere Göre Görevlerim</a></li>
                            <li><a href="#">Tamamlanmış Görevler</a></li>
                            <li><a href="#">Takvimlenmemiş Görevler</a></li>
                            <li role="separator" class="divider"></li>
                        </ul>
                    </li>

                    <li class="dropdown"><a href="#" class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="false">Kategorilerim<span
class="caret"></span></a>
                        <ul class="dropdown-menu">
                            <li><a href="category-all.php">Tüm Kategoriler</a></li>
                            <li role="separator" class="divider"></li>
                            <li><a href="category-edit.php">Kategori Ekle-Düzenle-Sil</a></li>
                        </ul>
                    </li>

                    <li class="dropdown"><a href="#" class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="false">Yardım<span class="caret"></span></a>
                        <ul class="dropdown-menu">
                            <li><a href="..../help/0_help_for_pertaskweb.html"
target="_blank">Çevrimiçi Yardım</a></li>
                            <li role="separator" class="divider"></li>
                            <li><a href="about.php" target="_blank">Hakkında</a></li>
                        </ul>
                    </li>
                </ul>
            </div>

            <ul class="nav navbar-nav navbar-right">
                <li class="dropdown"><a href="#" class="dropdown-toggle" data-toggle="dropdown"
role="button" aria-haspopup="true" aria-expanded="false">Profilim<span
class="caret"></span></a>
                    <ul class="dropdown-menu">
                        <li><a href="user-profile.php">Profilimi Göster</a></li>
                        <li role="separator" class="divider"></li>
                        <li><a href="change-password.php">Şifremi Değiştir</a></li>
                    </ul>
                </li>
                <li><a href="..../register/login_form.php" target="_blank">Çıkış</a></li>
            </ul>
        </div>
    </div>
</body>
```



```
</ul>

</div>
<!-- /.navbar-collapse -->
<!-- ===== -->

</div>
</div>

<div class="container" style="min-height:500px;">
    <div class=' '>
</div>

<script src="../js/jquery-1.11.3.min.js" type="text/javascript"></script>
<script src="../js/bootstrap.min.js" type="text/javascript"></script>
```

B.2.2 index.php (main_page)

```
<?php
// Developer: Group FEB
ob_start(); // "Warning: Cannot modify header information - headers already sent" hatasini
host ortamında önlemek için gerekiyor
require_once('bdd.php');
include('menu.php');

// time-date picker: http://jonthornton.github.io/jquery-timepicker/
session start();
$user = isset($_SESSION['user_name']) ? $_SESSION['user_name'] : '';
$logged_in = isset($_SESSION['logged_in']) ? $_SESSION['logged_in'] : '';

if ($logged_in != "yes"){
    header ("Location: ../register/login_form.php");
}

$sql = "SELECT id, title, start, end, cid, category FROM events WHERE user ='$user'";

$req = $bdd->prepare($sql);
$req->execute();

$events = $req->fetchAll();
ob_end_flush(); // "Warning: Cannot modify header information - headers already sent" hatasını
host ortamında önlemek için gerekiyor
?>

<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
    <link rel="icon" href="images/pertaskweb_icon.png">
    <link href="css/bootstrap.min.css" rel="stylesheet" type="text/css">

    <title>PerTaskWeb</title>

    <!-- Bootstrap Core CSS -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- FullCalendar -->
    <link href='css/fullcalendar.css' rel='stylesheet' />
    <link href='css/scheduler.min.css' rel='stylesheet' />
    <link href='css/fullcalendar.print.min.css' rel='stylesheet' media='print' />

    <!-- Custom CSS -->
    <style>
        body {
            /*padding-top: 70px; */
            padding-top: 0px;
```



```
        }
    #calendar {
        max-width: 800px;
    }
.col-centered{
    float: none;
    margin: 0 auto;
}
</style>
</head>

<body>
<!-- Page Content --&gt;
&lt;div class="container"&gt;

    &lt;div class="row"&gt;
        &lt;div class="col-lg-12 text-center"&gt;
            &lt;h1&gt;Takvimde Kişisel Görevlerim&lt;/h1&gt;
            &lt;div id="calendar" class="col-centered"&gt;
            &lt;/div&gt;
        &lt;/div&gt;
    &lt;/div&gt;

    &lt;/div&gt;

    &lt;!-- Modal: https://www.opensourcevarsity.com/b3-modal-tutorial/ --&gt;
    &lt;!-- Modal : https://getbootstrap.com/docs/4.0/components/modal/ --&gt;
    &lt;div class="modal fade" id="ModalAdd" tabindex="-1" role="dialog" aria-labelledby="myModalLabel"&gt;
        &lt;div class="modal-dialog" role="document"&gt;
            &lt;div class="modal-content"&gt;
                &lt;form class="form-horizontal" method="POST" action="addEvent.php"&gt;

                    &lt;div class="modal-header"&gt;
                        &lt;button type="button" class="close" data-dismiss="modal" aria-label="Close"&gt;&lt;span aria-hidden="true"&gt;&amp;times;&lt;/span&gt;&lt;/button&gt;
                        &lt;h4 class="modal-title" id="myModalLabel"&gt;Görev Ekle&lt;/h4&gt;
                    &lt;/div&gt;
                    &lt;div class="modal-body"&gt;

                        &lt;div class="form-group"&gt;
                            &lt;label for="title" class="col-sm-2 control-label"&gt;Görev&lt;/label&gt;
                            &lt;div class="col-sm-10"&gt;
                                &lt;input type="text" name="title" class="form-control" id="title" placeholder="Görev Adı"&gt;
                            &lt;/div&gt;
                        &lt;/div&gt;

                        &lt;div class="form-group"&gt;
                            &lt;label for="color" class="col-sm-2 control-label"&gt;Kategori&lt;/label&gt;
                            &lt;div class="col-sm-10"&gt;
                                &lt;?php
                                    //include("bdd_db.php");
                                    $sql = "SELECT * FROM categories WHERE user ='$user'";
                                    echo "&lt;select name=\"color\" class=\"form-control\" id=\"color\"&gt;";
                                    echo "&lt;option value=\"\"&gt;Başka Kategori Seçin&lt;/option&gt;";
                                    foreach ($bdd-&gt;query( $sql ) as $row) {
                                        echo "&lt;option style=\"color:" , $row['color'] , ";\" value=\"" , $row['category'] , " -" , $row['cid'] , "\">>";
                                        echo "&#9724; ";
                                        echo $row['category'];
                                        echo "</option>";
                                    }
                                    echo "</select>";
                                ?>

                            </div>
                        </div>

                        <div class="form-group">
                            <label for="start" class="col-sm-2 control-label">Başlangıç</label>
                            <div class="col-sm-10">
                                <input type="text" name="start" class="form-control" id="start" readonly>
                            </div>
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>
```



```
</div>
</div>
<div class="form-group">
    <label for="end" class="col-sm-2 control-label">Bitiş</label>
    <div class="col-sm-10">
        <input type="text" name="end" class="form-control" id="end" readonly>
    </div>
</div>

<!-- Bootstrap Modal for fullcalendar resources--&gt;

&lt;div class="form-group"&gt;
    &lt;!--
    &lt;label for="resourceid" class="col-sm-2 control-label"&gt;Kategori No&lt;/label&gt;
    --&gt;
    &lt;div class="col-sm-10"&gt;
        &lt;input type="hidden" name="resourceid" class="form-control" id="resourceid" readonly&gt;
    &lt;/div&gt;
&lt;/div&gt;

&lt;!-- end of model --&gt;

&lt;/div&gt;
&lt;div class="modal-footer"&gt;
    &lt;button type="button" class="btn btn-default" data-dismiss="modal"&gt;Kapat&lt;/button&gt;
    &lt;button type="submit" class="btn btn-primary"&gt;Kaydet&lt;/button&gt;
&lt;/div&gt;
&lt;/form&gt;
&lt;/div&gt;
&lt;/div&gt;
&lt;/div&gt;

<!-- Modal --&gt;
&lt;!-- Modal --&gt;
&lt;div class="modal fade" id="ModalEdit" tabindex="-1" role="dialog" aria-labelledby="myModalLabel"&gt;
    &lt;div class="modal-dialog" role="document"&gt;
        &lt;div class="modal-content"&gt;
            &lt;form class="form-horizontal" method="POST" action="editEventTitle.php"&gt;
                &lt;div class="modal-header"&gt;
                    &lt;button type="button" class="close" data-dismiss="modal" aria-label="Close"&gt;&lt;span aria-hidden="true"&gt;&amp;times;&lt;/span&gt;&lt;/button&gt;
                    &lt;h4 class="modal-title" id="myModalLabel"&gt;Görev Düzenle&lt;/h4&gt;
                &lt;/div&gt;
                &lt;div class="modal-body"&gt;

                    &lt;div class="form-group"&gt;
                        &lt;label for="title" class="col-sm-2 control-label"&gt;Görev&lt;/label&gt;
                        &lt;div class="col-sm-10"&gt;
                            &lt;input type="text" name="title" class="form-control" id="title" placeholder="Görev Adı"&gt;
                        &lt;/div&gt;
                    &lt;/div&gt;

                    &lt;div class="form-group"&gt;
                        &lt;label for="color" class="col-sm-2 control-label"&gt;Kategori&lt;/label&gt;
                        &lt;div class="col-sm-10"&gt;
                            &lt;?php
                                //include("bdd_db.php");
                                $sql = "SELECT * FROM categories WHERE user ='$user'";
                                echo "&lt;select name=\"color\" class=\"form-control\" id=\"color\"&gt;";
                                echo "&lt;option value=\"\"&gt;Başka Kategori Seçin&lt;/option&gt;";
                                foreach ($bdd-&gt;query( $sql ) as $row) {
                                    echo "&lt;option style=\"color:" . $row['color'] . ";\" value=\"" . $row['category'] . "\"&gt;" . $row['category'] . "&lt;/option&gt;";
                                }
                                echo "&lt;/select&gt;";
                            ?&gt;
                        &lt;/div&gt;
                    &lt;/div&gt;
                &lt;/div&gt;
            &lt;/form&gt;
        &lt;/div&gt;
    &lt;/div&gt;
&lt;/div&gt;</pre>
```



```
</div>

<!-- Bootstrap Modal for fullcalendar resources-->

<div class="form-group">
    <!--
    <label for="resourceid" class="col-sm-2 control-label">Kategori No</label>
    -->
    <div class="col-sm-10">
        <input type="hidden" name="resourceid" class="form-control" id="resourceid"
readonly>
    </div>
</div>

<!-- end of model -->

<div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
        <div class="checkbox">
            <label class="text-danger"><input type="checkbox" name="delete"> Görevi
Sil </label>
        </div>
    </div>
</div>

<input type="hidden" name="id" class="form-control" id="id">

</div>
<div class="modal-footer">
    <button type="button" class="btn btn-default" data-dismiss="modal">Kapat</button>
    <button type="submit" class="btn btn-primary">Kaydet</button>
</div>
</form>
</div>
</div>
</div>

</div>
<!-- /.container -->

<!-- jQuery Version 1.11.1 -->
<script src="js/jquery.js"></script>
<script src="js/jquery-ui.min.js"></script>

<!-- Bootstrap Core JavaScript -->
<script src="js/bootstrap.min.js"></script>

<!-- FullCalendar -->
<script src='js/moment.min.js'></script>
<script src='js/fullcalendar.js'></script>
<script src='js/scheduler.js'></script>
<script src='js/locale-all.js'></script>

<script>
$(document).ready(function() {

    $('#calendar').fullCalendar({
        locale: 'tr',
        schedulerLicenseKey: "GPL-My-Project-Is-Open-Source",
        header: {
            left: 'prev,next today',
            center: 'title',
            right: 'timelineDay, timelineThreeDays, listMonth'
        },
        views: {
            timelineDay: {
                type: 'timeline',
                titleFormat: 'ddd D MMMM YYYY',
                slotDuration: '00:30:00',
                slotLabelFormat: 'HH:mm',
                duration: { days: 1 }
            },

```

```

        timelineThreeDays: {
            type: 'timeline',
            titleFormat: 'ddd D MMMM YYYY',
            slotDuration: '00:30:00',
            slotLabelFormat: 'HH:mm',
            duration: { days: 3 }
        },
    },
    defaultView: 'timelineDay',
    resourceAreaWidth: '30%',
    resourceColumns: [
        {
            labelText: 'KATEGORİ',
            field: 'kategori',
        },
        //{
        //    labelText: 'RENK',
        //    field: 'renk'
        //}
    ],
    resources: 'category_get.php', // veritabanından okuyup otomatik resource yükleme
    resourceAreaWidth: "30%",

    defaultDate: '2019-01-02',
    editable: true,
    eventLimit: true, // allow "more" link when too many events
    selectable: true,
    selectHelper: true,
    scrollTime: '00:00',
    select: function(start, end, jsEvent, view,resource) { //when the cell(s) selected
the textboxes start and end
        // in modal ModalAdd are filled
        // according to the selected start cell and end cell:
        //alert(resource.id);
    },
    eventRender: function(event, element) { //when an event on the calendar double
clicked
        // the properties of that event are
transferred to modal "ModelEdit" 's properties
        // and finally the modal "ModelEdit" is
shown with the properties transferred:
        element.bind('dblclick', function() {
            $('#ModalEdit #id').val(event.id);
            $('#ModalEdit #title').val(event.title);
            $('#ModalEdit #resourceid').val(event.resourceId);
            $('#ModalEdit').modal('show'); //Manually opens the modal
        });
    },
    eventDrop: function(event, delta, revertFunc) { // konumu (tarihi) değiştirme
edit(event);
},
    eventResize: function(event, dayDelta, minuteDelta, revertFunc) { // uzunluğunu
değiştirme
edit(event);
},
    events: [
<?php foreach($events as $event):

```

```

$end = explode(" ", $event['end']);
if($start[1] == '00:00:00'){
    $start = $start[0];
} else{
    $start = $event['start'];
}
if($end[1] == '00:00:00'){
    $end = $end[0];
} else{
    $end = $event['end'];
}
$cid = $event['cid'];

?>
{
    id: '<?php echo $event['id']; ?>',
    resourceId: '<?php echo $event['cid']; ?>',
    title: '<?php echo $event['title']; ?>',
    start: '<?php echo $start; ?>',
    end: '<?php echo $end; ?>',
},
<?php endforeach; ?>
]
});

function edit(event){
    start = event.start.format('YYYY-MM-DD HH:mm:ss');
    if(event.end){
        end = event.end.format('YYYY-MM-DD HH:mm:ss');
    }else{
        end = start;
    }
    id = event.id;
    resourceId = event.resourceId;
    Event = [];
    Event[0] = id;
    Event[1] = start;
    Event[2] = end;
    Event[3] = resourceId;
    $.ajax({ //this ajax method sends the properties of the event selected on the
calendar to the editEventDate.php:
        // read: http://www.aydinmahmut.com/jquery-ajax-ile-post-ve-get-islemeleri/
        url: 'editEventDate.php',
        type: "POST",
        data: {Event:Event}, // calendar'dan okunan sagdaki Event verileri soldakine
aktarilir
        // ve soldaki Event objesi, veritabanina kaydedilmek
        // üzere editEventDate.php'ye gonderilir
        success: function(taken) { //success yani gönderme başarılı olursa çalışacak
fonksiyondur.
            // buradaki taken değişkeni değeri ditEventDate.php
            içindeki die('x') ile dönen x değerdir.
            if(taken == 'OK'){
                //alert('Kaydedildi...');

            }else{
                alert(taken);
                //alert('Kaydedilemedi, tekrar deneyin...');

            }
        },
        error:function(code,info)
        {
            if (code.status === 0) {
                alert('Internet Bağlantısı Yok...');
            } else if (code.status == 404) {
                alert('İstenen Sayfa Bulunamadı... [Error No: 404]');
            } else if (code.status == 500) {
                alert('Sunucu Hatası... [Error No: 500].');
            } else if (info === 'timeout') {
                alert('Zaman Aşımı Hatası...');

            } else if (info === 'abort') {
                alert('Ajax request rejected.');

            } else {
                alert('Bir hata oluştu...');

            }
        }
    })
}

```



```
        alert('Tanimlanmamis Hata...\\n' + code.responseText);
    }
}
});

});
</script>
</body>
</html>
```

B.2.3 addEvent.php

```
<?php
// Developer: Group FEB
// Database connection:
require_once('bdd.php');
session_start();
$user = isset($_SESSION['user_name']) ? $_SESSION['user_name'] : '';
//echo $ POST['title'];
if (isset($_POST['title']) && isset($_POST['start']) && isset($_POST['end']) &&
isset($_POST['resourceid'])) {

$title = $_POST['title'];
$start = $_POST['start'];
$end = $_POST['end'];
$resourceid = $_POST['resourceid'];

if (isset($_POST['color']) AND ($_POST['color']) != "") {
$color_category = $_POST['color'];
$colorcategory = explode("-", $color_category);
$category = $colorcategory[0];
$cid = $colorcategory[1];
} else {
$cid = $resourceid;
$sql = "SELECT * FROM categories WHERE user = '$user' AND cid = '$cid' ";
foreach ($bdd->query($sql) as $row) {
$category = $row['category'];
}
}

$sql = "INSERT INTO events(title, start, end, user, category, cid) values ('$title',
'$start', '$end', '$user', '$category', $cid )";

$query = $bdd->prepare($sql);
if ($query == false) {
print_r($bdd->errorInfo());
die ('Hata');
}
$stmt = $query->execute();
if ($stmt == false) {
print_r($query->errorInfo());
die ('Hata');
}
}
header('Location: '.$_SERVER['HTTP_REFERER']);
?>
```



B.2.4 addEventTitle.php

```
<?php
// Developer: Group FEB
require_once('bdd.php');
if (isset($_POST['delete']) && isset($_POST['id'])){

    $id = $_POST['id'];

    $sql = "DELETE FROM events WHERE id = $id";
    $query = $bdd->prepare( $sql );
    if ($query == false) {
        print_r($bdd->errorInfo());
        die ('Sorgu Hazırlamada Hata var!');
    }
    $res = $query->execute();
    if ($res == false) {
        print_r($query->errorInfo());
        die ('Sorgu Çalıştırımda Hata Var!');
    }
} elseif (isset($_POST['title']) && isset($_POST['resourceid']) && isset($_POST['id'])) {
    $id = $_POST['id'];
    $title = $_POST['title'];
    // $color = $_POST['color'];

    $resourceid = $_POST['resourceid'];

    if (isset($_POST['color']) AND ($_POST['color']) != "") {
        $color_category = $_POST['color'];
        $colorcategory = explode("-", $color_category);
        $category = $colorcategory[0];
        $cid = $colorcategory[1];
        $sql = "UPDATE events SET title = '$title', category = '$category', cid = '$cid' WHERE
    id = $id ";
    } else {
        $cid = $resourceid;
        $sql = "UPDATE events SET title = '$title' WHERE id = $id ";
    }

    $query = $bdd->prepare( $sql );
    if ($query == false) {
        print_r($bdd->errorInfo());
        die ('Sorgu Çalıştırımda Hata Var!');
    }
    $sth = $query->execute();
    if ($sth == false) {
        print_r($query->errorInfo());
        die ('Sorgu Çalıştırımda Hata Var!');
    }
}
header('Location: index.php');
?>
```

B.2.5 editEventDate.php

```
<?php
// Developer: Group FEB
// Veritabanına bağlantı:
session_start();
require_once('bdd.php');
$user = isset($_SESSION['user_name']) ? $_SESSION['user_name'] : '';
// ajax'dan gönderilen Event objesinin elemanları burada kullanılır:
// $_POST is the global variable to access values sent by Jquery's ajax method
if (isset($_POST['Event'][0]) && isset($_POST['Event'][1]) && isset($_POST['Event'][2]) &&
    isset($_POST['Event'][3])) {
    $id = $_POST['Event'][0];
    $start = $_POST['Event'][1];
    $end = $_POST['Event'][2];
    $resourceid = $_POST['Event'][3];
    $cid = $resourceid;

    $sql2 = "SELECT * FROM categories WHERE user = '$user' AND cid = $cid" ;
```



```
foreach ($bdd->query( $sql2 ) as $row) {
    $category = $row['category'];
}

$sql = "UPDATE events SET start = '$start', end = '$end', cid = '$cid', category =
'$category' WHERE id = $id ";

$query = $bdd->prepare( $sql );
if ($query == false) {
    print_r($bdd->errorInfo());
    die ('Hata: Sorgu çalışmadı....');
}
$sth = $query->execute();
if ($sth == false) {
    print_r($query->errorInfo());
    die ('ERROR'); // 'ERROR' ifadesi bu php'yi çağırılan ajax içindeki success: function(x)
içindeki x değerine gönderilir
} else{
    die ('OK'); // 'OK' ifadesi bu php'yi çağırılan ajax içindeki success: function(x)
içindeki x değerine gönderilir
}

}

?>
```

B.3. Category Codes

B.3.1 category.php

```
<?php
// Developer: Group FEB
session_start();
include('menu.php');
include('header.php');
include_once("connect_db.php");

$stab_menu = '';
$stab_content = '';
$user = isset($_SESSION['user_name']) ? $_SESSION['user_name'] : '';
$i = 0;

$sql = "SELECT * FROM categories WHERE user = '$user' ORDER BY cid ASC";

foreach ($connect_db->query( $sql ) as $row) {
    if($i == 0)
    {
        $stab_menu .= '
            <li class="active"><a href="#'.$row["cid"].'" data-
        toggle="tab">' . $row["category"] . '</a></li>
        ';
        $stab_content .= '
            <div id="'.$row["cid"].'" class="tab-pane fade in active">
        ';
    }
    else
    {
        $stab_menu .= '
            <li><a href="#'.$row["cid"].'" data-toggle="tab">' . $row["category"] . '</a></li>
        ';
        $stab_content .= '
            <div id="'.$row["cid"].'" class="tab-pane fade">
        ';
    }
}

$task_sql = "SELECT * FROM events WHERE user = '$user' AND category =
'". $row["category"] . "'";
$stab_content .= '
    <table width="300" border="1">
        <tbody>
            <tr>
                <th width="100" scope="col">' . 'TASK' . '</th>
                <th width="100" scope="col">' . 'START-DATE' . '</th>
                <th width="100" scope="col">' . 'END-DATE' . '</th>
```



```
</tr>
';

foreach ($connect_db->query( $task_sql ) as $sub_row) {

    $tab_content .= '
        <tr>
            <td>' . $sub_row["title"] . '</td>
            <td>' . $sub_row["start"] . '</td>
            <td>' . $sub_row["end"] . '</td>
        </tr>
    ';
}

$tab_content .= '
    </tbody>
</table>
';

$tab_content .= '<div style="clear:both"></div></div>';

$i++;
}

?>

<!DOCTYPE html>
<html>
<head>
<title>Kategorilerime göre Görevlerim</title>
<script src="js/jquery.js"></script>
<script src="js/bootstrap.min.js"></script>
<link rel="stylesheet" href="css/bootstrap.min.css" />
</head>
<body>
<div class="container">
    <h2 align="center">Kategorilerime göre Görevlerim</a></h2>
    <br />
    <ul class="nav nav-tabs">
        <?php echo $tab_menu; ?>
    </ul>
    <div class="tab-content">
        <br />
        <?php echo $tab_content; ?>
    </div>

    <br />
    <div>
        <button type="submit" class="btn btn-default" name="btn-exit" id="btn-exit"
value="Çık" onClick="location.href='index.php'" >
            <span class="glyphicon glyphicon-log-in"></span> &ampnbsp Çık
        </button>
    </div>
</body>
</html>
```

B.3.2 category-edit.php

```
<?php
// Developer: Group FEB
session_start();
include('menu.php');
include('header.php');
include_once("connect_db.php");

$searchedId = "";
$searchedCategory = "";
$searchedColor = "2111CE";
$searchedTextColor = "FFFEBO";

$user = isset($_SESSION['user_name']) ? $_SESSION['user_name'] : '';
```



```
//add function:  
if ( isset($_POST['addcategory'])) {  
  
    $category = $_POST['category'];  
    $color = "#" . $_POST['color'];  
    $textcolor = "#" . $_POST['textcolor'];  
    if ( isset($_POST['parent'])) {  
        $parentid = $_POST['parent'];  
    } else {  
        $parentid = NULL;  
    }  
    $user = $user;  
  
    //Let's remove unwanted HTML (scripting attacks):  
    $category = htmlspecialchars($category);  
  
    $sql = "SELECT * FROM categories WHERE user='$user'";  
  
    $query = $connect_db->prepare( $sql );  
    $sth = $query->execute();  
    $rows = $query->fetchAll(PDO::FETCH_ASSOC);  
  
    foreach($rows as $row) {  
        if($row['category'] == $category){  
            echo "Bu kategori zaten var";  
            exit();  
        }  
    }  
  
    $sql = "INSERT INTO categories(category, color, textcolor, parentid, user) VALUES  
('{$category}', '{$color}', '{$textcolor}', '$parentid', '$user' )";  
    $query = $connect_db->prepare( $sql );  
    $sth = $query->execute();  
}  
  
//delete function:  
if ( isset($_POST['deletecategory'])) {  
    $cid = $_POST['cid'];  
    $sql = "DELETE FROM categories WHERE cid = '$cid' OR parentid = '$cid' ";  
    $query = $connect_db->prepare( $sql );  
    if ($query == false) {  
        print_r($connect_db->errorInfo());  
        die ('Sorgu Hazırlamada Hata var!');  
    }  
    $res = $query->execute();  
    if ($res == false) {  
        print_r($query->errorInfo());  
        die ('Sorgu Çalıştırımda Hata Var!');  
    }  
  
    $sql = "DELETE FROM events WHERE user = '$user' AND cid = '$cid' ";  
    $query = $connect_db->prepare( $sql );  
    if ($query == false) {  
        print_r($connect_db->errorInfo());  
        die ('Sorgu Hazırlamada Hata var!');  
    }  
    $res = $query->execute();  
    if ($res == false) {  
        print_r($query->errorInfo());  
        die ('Sorgu Çalıştırımda Hata Var!');  
    }  
  
}  
  
//update function:  
if ( isset($_POST['updatecategory'])) {  
    $category = $_POST['stdcategory'];  
    $color = "#" . $_POST['stdcolor'];  
    $textcolor = "#" . $_POST['stdtextcolor'];  
    $cid = $_POST['stdid'];  
    $user = $user;  
  
    $sql = "UPDATE categories SET category = '$category', color = '$color',  
textcolor = '$textcolor' WHERE user = '$user' AND cid = $cid ";  
    $query = $connect_db->prepare( $sql );
```



```
if ($query == false) {
    print_r($$connect_db->errorInfo());
    die ('Sorgu Çalıştırmada Hata Var!');
}
$sth = $query->execute();
if ($sth == false) {
    print_r($query->errorInfo());
    die ('Sorgu Çalıştırmada Hata Var!');
}

$sql = "UPDATE events SET category = '$category' WHERE user = '$user' AND cid = $cid";
$query = $connect_db->prepare( $sql );
if ($query == false) {
    print_r($$connect_db->errorInfo());
    die ('Sorgu Çalıştırmada Hata Var!');
}
$sth = $query->execute();
if ($sth == false) {
    print_r($query->errorInfo());
    die ('Sorgu Çalıştırmada Hata Var!');
}

}

if ( isset($_POST['searchcategory']) ){
    $searchCategory = $_POST['category'];
    $sql = "SELECT * FROM categories WHERE user = '$user' AND category = '$searchCategory' ";

    foreach ($connect_db->query( $sql ) as $row) {
        $searchedCategory = $row['category'];
        $searchedColor = $row['color'];
        $searchedTextColor = $row['textcolor'];
        $searchedId = $row['cid'];
    }
}
?>
<html>
    <head>
        <script type="text/javascript" src="script/jscolor.js"></script>
        <link href="css/style.css" rel="stylesheet" type="text/css" media="screen">
        <link href="css/bootstrap.min.css" rel="stylesheet">
        <link href="css/product.css" rel="stylesheet">
    </head>
    <body>
        <h3><strong>Kategori Ekle</strong></h3>
        <form name="addition" action="category-edit.php" method="post">
            <label>Üst Kategori:</label>
            <?php
                $sql = "SELECT * FROM categories WHERE user = '$user'";
                echo "<select name=\"parent\">";
                echo "<option value=\"\">Üst kategori seç</option>";
                foreach ($connect_db->query( $sql ) as $row) {
                    echo "<option style=\"color:{$row['color']}\" value=\"$row['cid']\">";
                    echo "&#9724; ";
                    echo $row['category'];
                    echo "</option>";
                }
                echo "</select>";
            ?>

            <input type="text" class="form-control" name="category" placeholder="Kategori Adı" required="required" /><br />
            <label>Kategori Rengi:</label>
            <input type="text" class="jscolor" name="color" size="7" placeholder="Renk Seçin" required="required" value="<?php echo '#FF0000'; ?>"/> &nbsp;
            <label>Yazı Rengi:</label>
            <input type="text" class="jscolor" name="textcolor" size="7"
```



```
placeholder="Renk Seçin" required="required" value="<?php echo '#FFF69A'; ?>"/> &nbsp;

<button type="submit" class="btn btn-default" name="addcategory" value="Kaydet">
    <span class="glyphicon glyphicon-log-in"></span> &nbsp; Kaydet
</button>
</form>

<h3><strong>Kategori Sil</strong></h3>
<form name="deletion" action="category-edit.php" method="post">
    <label>Kategori Seç :</label>

    <?php

        $sql = "SELECT * FROM categories WHERE user = '$user'";
        echo "<select name=\"cid\" required>";
        echo "<option value=\"\">Kategori Seçin</option>";
        foreach ($connect_db->query( $sql ) as $row) {
            echo "<option style=\"color:{$row['color']}\" value=\"{$row['cid']}\">";
            echo "&#9724; ";
            echo $row['category'];
            echo "</option>";
        }
        echo "</select>";
    ?>

    <button type="submit" class="btn btn-default" name="deletecategory" value="Sil">
        <span class="glyphicon glyphicon-log-in"></span> &nbsp; Sil
    </button>
</form>

<h3><strong>Kategori Düzenle</strong></h3>
<form name="edition" action="category-edit.php" method="post">
    <label>Kategori Seç:</label>

    <?php

        $sql = "SELECT * FROM categories WHERE user = '$user'";
        echo "<select name=\"category\" required>";
        echo "<option value=\"\">Kategori Seçin</option>";
        foreach ($connect_db->query( $sql ) as $row) {
            echo "<option style=\"color:{$row['color']}\" value=\"{$row['category']}\">";
            echo "&#9724; ";
            echo $row['category'];
            echo "</option>";
        }
        echo "</select>";
    ?>
    <button type="submit" class="btn btn-default" name="searchcategory" value="Ara">
        <span class="glyphicon glyphicon-log-in"></span> &nbsp; Ara
    </button>
</form>

<form name="stdedit" action="category-edit.php" method="post">
    <input name="stdid" required="required" class="form-control" type="hidden" readonly value="<?php echo $searchedId; ?>" />
    <label>Kategori Adı:</label>
    <input name="stdcategory" required="required" class="form-control" type="text" value="<?php echo $searchedCategory; ?>" /><br />
    <label>Kategori Rengi:</label>
    <input type="text" class="jscolor" size="7" placeholder="Renk Seçin" name="stdcolor" required="required" value="<?php echo $searchedColor; ?>"/>
    <label>Yazı Rengi:</label>
    <input type="text" class="jscolor" name="stdtextcolor" size="7" placeholder="Renk Seçin" required="required" value="<?php echo $searchedTextColor; ?>"/>
    &nbsp;

    <button type="submit" class="btn btn-default" name="updatecategory" value="Güncelle">
        <span class="glyphicon glyphicon-log-in"></span> &nbsp; Güncelle
    </button>
</form>
```



```
<div>
    <button type="submit" class="btn btn-default" name="btn-exit" id="btn-submit"
value="Çık" onClick="location.href='index.php'" >
        <span class="glyphicon glyphicon-log-in"></span> &ampnbsp Çık
    </button>
</div>

</body>
</html>
```

B.4 User Profile Codes

B.4.1 user_profile.php

```
<?php
// Developer: Group FEB
session_start();
include('menu.php');
include('header.php');
include_once("bdd.php");
$user = isset($_SESSION['user_name']) ? $_SESSION['user_name'] : '';
$sql = "SELECT * FROM users WHERE user = '$user'";
$query = $bdd->query($sql);
$row = $query->fetch(PDO::FETCH_ASSOC);
$email = $row['email'];
$password = $row['pass'];
$firstname = $row['firstname'];
$surname = $row['surname'];
?>





```