System Design Document

PAT Phase 2 – Grade 12



Prepared by Inge Senekal

Curro Roodeplaat

14/04/2021

Contents

[1. User Interface Design 1](#_Toc76064119)

[1.1 Login Screen 1](#_Toc76064120)

[1.2 Register Screen 2](#_Toc76064121)

[1.3 Application Screen 3](#_Toc76064122)

[1.4 Task List Screen 4](#_Toc76064123)

[1.5 Task Detail Screen 5](#_Toc76064124)

[1.5 Planner Screen 6](#_Toc76064125)

[1.6 Planner Detail Screen 7](#_Toc76064126)

[1.7 Help Screen 8](#_Toc76064127)

[2. Sequencing 9](#_Toc76064128)

[2.1 Login Screen Algorithms 9](#_Toc76064129)

[2.2 Register Screen Algorithms 9](#_Toc76064130)

[2.3 Application Screen Algorithms 10](#_Toc76064131)

[2.4 Task List Screen Algorithms 10](#_Toc76064132)

[2.5 Task Detail Screen Algorithms 11](#_Toc76064133)

[2.6 Planner Screen Algorithms 12](#_Toc76064134)

[2.7 Planner Detail Screen Algorithms 12](#_Toc76064135)

[3. Class Design 14](#_Toc76064136)

[3.1 User Class 14](#_Toc76064137)

[3.2 Task Class 15](#_Toc76064138)

[4. Persistent Storage Design 16](#_Toc76064139)

[4.1 tblTasks 16](#_Toc76064140)

[4.1.1 tblTasks Sample Data 16](#_Toc76064141)

[4.2 tblSubjectPriority 17](#_Toc76064142)

[4.2.1 tblSubjectPriority Sample Data 17](#_Toc76064143)

[4.3 tblSubject 17](#_Toc76064144)

[4.3.1 tblSubject Sample Data 17](#_Toc76064145)

[4.4 tblTaskStatus 17](#_Toc76064146)

[4.4.1 tblTaskStatus Sample Data 17](#_Toc76064147)

[4.5 tblUser 18](#_Toc76064148)

[4.5.1 tblUser Sample Data 18](#_Toc76064149)

[5. Explanation of Storage Design 19](#_Toc76064150)

[5.1 Primary Storage 19](#_Toc76064151)

[5.1.1 User Class 19](#_Toc76064152)

[5.1.2 Task Class 19](#_Toc76064153)

[5.2 Secondary Storage 19](#_Toc76064154)

# User Interface Design

## Login Screen



|  |  |
| --- | --- |
| **Description** | Screen appear when program runs and will be used to login.  It contains text boxes for the user to enter their username and password into and a Register and Login button. |
| **Data** | The user’s username and password will be entered. |
| **Actions** | **Register**   * If button is pressed it will move to a register screen   **Login**   * Check if values are filled in. * If not, display an error message. * Check if user name exists. * If not, display an error message. * If it does * Check if the password is correct. * If not * Display an error message. * If valid * Display the Application screen. |

## Register Screen



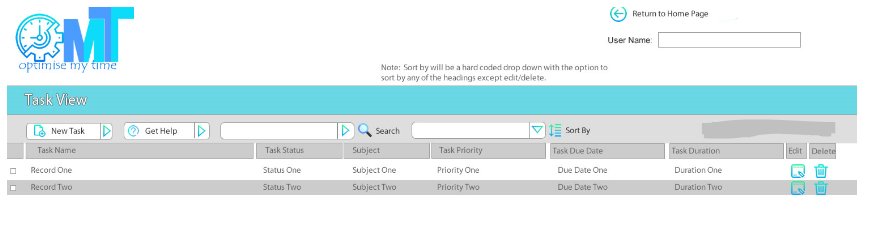
|  |  |
| --- | --- |
| **Description** | This screen will be used to Register  This allows a new user to create a new account  It contains text boxes for the user to enter their Full Name, Password, Telephone number and Email address into and a Register button.  Label provided for what to enter where and at the bottom of the text book a description of what kind of information is needed. |
| **Data** | The user’s Full name Password, Telephone number and Email Address will be entered. |
| **Actions** | **Full Name field:**   * Type in users name and surname.   **Password field:**   * Type in a password with a minimum of 8 characters.   **Telephone field:**   * Type a telephone number with exactly 10 numbers.   **Email Address:**   * Type in an email address.   **Register**  Check if values are filled in.   * If not, display an error message.   Check if user exists.   * If it does, display an error message.   If not   * Add the user to the database. * Display login screen |

## Application Screen



|  |  |
| --- | --- |
| **Description** | This screen will be used as the main screen.  It contains buttons to the users’ task list, calendar, help and logout. |
| **Data** | This screen only allows user to move to different screens.  No data will be stored here |
| **Actions** | **Task List**  If icon “task list” is pressed   * Display the Task List Screen   **Calendar**  If icon “calendar” is pressed   * Display the Planner Screen   **Help**  If “Get Help” is pressed   * Display the Help Screen   **Log out:**  If icon log out is pressed   * End program |

## Task List Screen



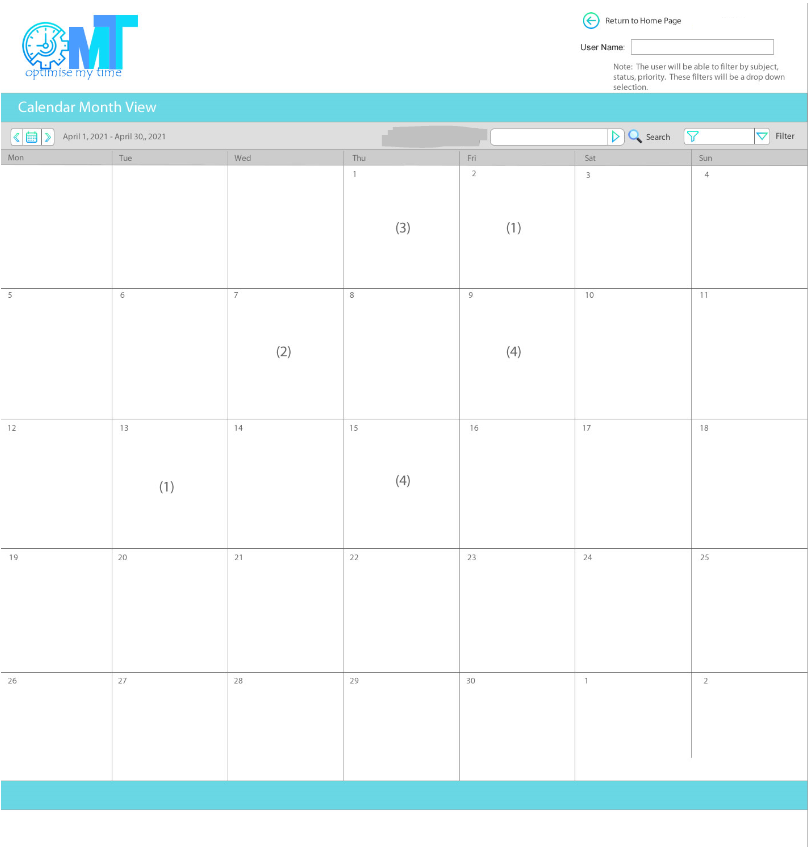
|  |  |
| --- | --- |
| **Description** | This form is used for the user to see all their task.  The user buttons are “New Task”, “ Get Help”, “Return to Home Page”, “Edit” and “Delete”. “Search” and “Sort By” will be a drop down allowing the user to decide on what variable they want to either search or sort by.  The information here will only be for view purposes so no data will be entered here.  The users name will also be displayed at the top of the screen. |
| **Data** | No data will be entered on this screen.  User can only view data or decide to edit that which will take then to a next screen.  Data can however be deleted on this screen. |
| **Actions** | **Return to Home Page Button**  If button is pressed   * Display the Application Screen   **New Task Button**  If button is pressed   * Display the Task Detail Screen   **Search Drop Down**  If pressed.   * show variables to choose from * show all records that the variable is found in.   **Sort by Drop Down**  If pressed.   * show variables to choose from * show variables in descending order.   **Edit Button**  If button is pressed   * Display the Task Detail Screen   **Delete Button**  If button is pressed   * Display the Application Screen |

## 1.5 Task Detail Screen



|  |  |
| --- | --- |
| **Description** | This screen will be used to edit or add a new task.  It contains text boxes for the user to enter their Task Name, Subject, Start and End date, Time Allocation and Task Description into.  It has a Submit button |
| **Data** | The user’s task will be entered, the subject the task is in, the start date end date, the time allocated and a short description of the task. |
| **Actions** | **Submit**  Check if values are filled in.   * If not, display an error message.   If valid   * display the Task List screen. |

## Planner Screen



|  |  |
| --- | --- |
| **Description** | This screen will be used to display the purpose of the program. It contains Drop downs for Search and Filter. |
| **Data** | No data will be entered all data will be displayed. This screen will consist of a calendar where the user can view how many tasks they have. Amount of task will be displayed through a number |
| **Actions** | **Each Day is a button**  If pressed.   * Display planner detail screen   **Calendar Icon**  If pressed.   * Shows month to choose from   **Search Drop Down**  If pressed.   * show variables to choose from * show variables in descending order.   **Filter Drop Down**  If pressed.   * show variables to choose from(subject, status, priority) * show variables in descending order. |

## Planner Detail Screen



|  |  |
| --- | --- |
| **Description** | This screen will be used to view the task information for the day chosen in the week. It consist of only stored information nothing for the user to input. Contain one button “Close” |
| **Data** | No data will be entered here however data will be the stored data. Data that will be displayed is, “Task Name”, “Subject”, Start and End Date”, “Time Allocation”, “Days Remaining”, “Task Description” |
| **Actions** | **Close**  If pressed:   * Display Planner Screen |

## Help Screen

|  |  |
| --- | --- |
| **Description** | This screen will be used to provide help for the user. It contains a drop box for the user to select what topic they need help with. There is only one button to allow you to go back to the Application Screen |
| **Data** | No data will be entered here all information will be hard coded in the program user can only decide what topic they need help with at the drop box. The help will be about each screen. And the features on each screen |
| **Actions** | **Return to Home Page Button**  If button is pressed   * Display the Application Screen   **Search Drop Down**  If pressed.   * show variables to choose from |



# Sequencing

* When program runs it will start at the login screen
* If the user ends the program the user will be taken to Login screen if the program runs again

## Login Screen Algorithms

* User inputs information after user will be taken to the Application Screen.
* But if a user does not have an account and wants to make one, they should click on the register button which will take them to the register screen.

Header: frmLogin

Begin

INPUT Username

INPUT Password

IF “Login” Button is pressed

IF Username field is empty OR (||) password field is empty

DISPLAY error message that values are required or register.

ELSE

IF Username exist

IF Password correct

DISPLAY Application Screen

ELSE

DISPLAY Error message username OR password is incorrect.

IF “Register” Button is pressed

DISPLAY Register Screen

END

## Register Screen Algorithms

* This screen is only accessed when user makes a new account and requires the user to enter personal information, when information is valid the user will be taken to the Application screen.

Header: frmRegister

BEGIN

INPUT Username

INPUT Name

INPUT Telephone Number

INPUT Password

IF “Submit” Button Pressed

IF Any field is empty

DISPLAY error message that values are required.

ELSE

IF username exists

DISPLAY error message that the username is not unique.

ELSE

Add user details to the database.

DISPLAY Login Screen

END

## Application Screen Algorithms

* This is the main screen which allows users to move between the different main screen. This screen has no data entering requirements and if the user exits from any other screen, they will be taken to this screen.
* This is the only screen that allows the program to end.

Header: frmApplication

BEGIN

DISPLAY Username

IF “Task List” Button is pressed

DISPLAY Task List Screen

IF “Planner” Button is pressed

DISPLAY Planner Screen

IF “Help” Button is pressed

DISPLAY Help Screen

IF “Log out” Button is pressed

END program

END

## Task List Screen Algorithms

* User moves to this screen if button was clicked “Task List” button was clicked on Application screen.
* It displays all tasks and information on each task.
* This screen allows the user to make changes to the database or just view information entered in database.
* If the user is new no information will be on this screen and user will have to enter new task
* If a user decides to edit, view or add a new task they will be taken to the “Task Detail Screen”.

Header: frmTaskView

BEGIN

IF “Return to Home Page” Button is pressed

DISPLAY Application Screen

DISPLAY Username

DISPLAY Task Name <tab> Task Status <tab> Task Priority <tab> Task Due Date <tab> Task Duration

IF “New Task” Button is pressed

DISPLAY Task Detail Screen

IF “SEARCH” Drop Down Pressed

Give User Option on which variable to Search.

DISPLAY Task Details Searched for

IF “Sort by” Drop Down Pressed

Give User Option on which variable to Sort.

Display information in DESC order

IF “Edit” Button is pressed

DISPLAY Task Detail Screen

IF “Delete” Is pressed

Task must be removed from screen.

END

## Task Detail Screen Algorithms

* This screen is accessed when a user decides to view, edit or add a new task to the database.
* If the user is finished with this screen, they will be taken back to the “Task List Screen”.

Header: frmTaskDetail

BEGIN

IF “New Task”

INPUT Task Name

INPUT Related Subject

INPUT Start Date

INPUT End Date

INPUT Time Allocation

INPUT Task Description

IF “Submit” Button is pressed

IF Any field is empty

DISPLAY error message that values are required.

ELSE

DISPLAY Task List Screen with new record listed.

IF “Edit”

INPUT Where variable has to change.

IF Any field is empty

DISPLAY error message that values are required.

ELSE

DISPLAY Task List Screen with edit record fixed.

END

## Planner Screen Algorithms

* User moves to this screen if button was clicked “Planner” button was clicked on Application screen.
* Here the viewer can see task planned out on a calendar.
* If user want to see task for a specific day the user must click on the day and it will take the user to planner detail screen
* No information can be altered on this screen if a user is not satisfied with a task they must go back to the “Task List Screen”

Header: frmPlanner

BEGIN

IF “Return to Home Page” Button pressed

DISPLAY Application Screen

DISPLAY Username

IF “Search” Button

Show variables to choose from

DISPLAY Task with matched variable

IF “Filter” Drop Down

Show variables to choose from

DISPLAY variable in DESC order

IF ANY “Day” is selected

DISPLAY Planner Detail Screen

END

## Planner Detail Screen Algorithms

* This screen will be displayed when the user decides to view a task for a specific day.
* No data can be entered it is only for viewing purposes.
* If the user exits this screen, it will go back to the “Planner Screen”.

Header: frmPlannerDetail

BEGIN

DISPLAY Task Name

DISPLAY Related Subject

DISPLAY Start Date

DISPLAY End Date

DISPLAY Time Allocation

DISPLAY Days Remaining

DISPLAY Task Description

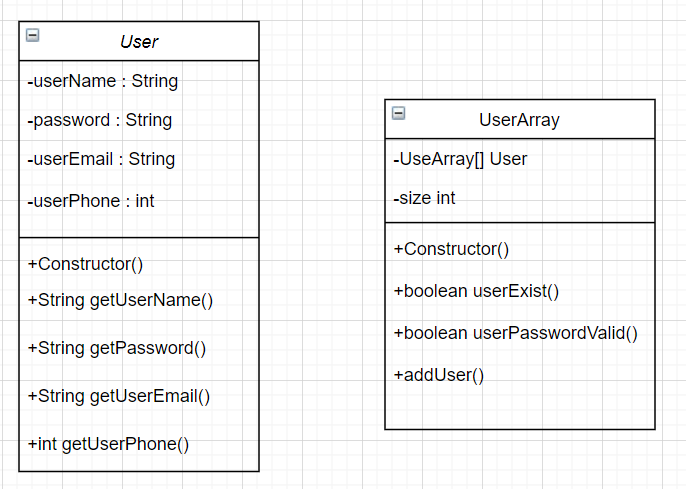
IF “Close” Button Pressed

DISPLAY Planner Screen

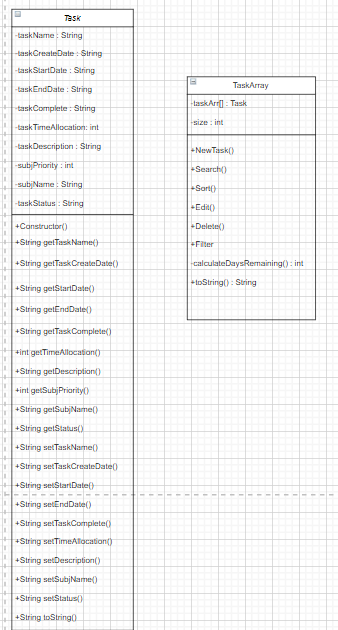
END

# Class Design

## User Class



## Task Class



# Persistent Storage Design

The database diagram includes the database tables with all details related to each table including:

* Table names
* Primary and Foreign keys
* Fields – names and types
* Sample data

//If I have more time

## tblTasks

This is the main table for storing information related to the tasks entered by the user.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary Key | Foreign Key | Field Name | Types | Description |
| TaskID | tblTasks\_tblSubject (TaskSubjectID-SubjectID)  tlbTasks\_tblSubjectPriority  (TaskSubjProirityID-SubjPriorityID)  tblTasks\_tblTaskStatus  (TaskStatusID-TaskStatusID)  tblTasks\_tblUser  (TaskUserID-UserID) | TaskID  TaskName  TaskUserID  TaskStatusID  TaskSubjectID  TaskSubjPriorityID  TaskCreateDate  TaskStartDate  TaskDueDate  TaskComplete  TaskCompleteDate  TaskTimeAllocation  TaskDescription  //TaskInactive\* | AutoNumber  Short Text  AutoNumber  AutoNumber  AutoNumber  AutoNumber  Date/ Time  Date/ Time  Date/ Time  True/False  Date/ Time  Number/Integer  Long Text  //True/ False | System assigned unique identifier.  Name of task  Unique ID of user  Unique ID of status  Unique ID of subject  Unique ID of subject priority  System timestamp for date/time task created.  Start date assigned.  Due date assigned.  Tick box  Date task completed.  Time allocated (minutes)  Description of task  //Flag for when task has been deleted by user |

### tblTasks Sample Data

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TaskID** | **Task**  **Name** | **User ID** | **Status ID** | **Subject ID** | **TaskSubj**  **Priority ID** | **Task**  **Create Date** | **Task**  **StartDate** | **Task**  **DueDate** | **Task**  **Complete** | **Task**  **Complete**  **Date** | **Task Time**  **Allocation** | **Task**  **Description** | **//Task**  **Inactive** |
| 1 | Mid-year CAT task | 1 | 1 | 1 | 1 | 26/04/2021  14:40:02 | 27/04/2021 | 04/05/2021 | Yes | 27/04/2021 | 80 | Mid-year CAT  task as  per documentation | No |
| 2 | 1st term IT task | 2 | 2 | 2 | 2 | 21/04/2021  12:20:04 | 26/04/2021 | 09/05/2021 | No |  | 120 | 1st term IT task includes.  create mini.  reservation application | No |

## tblSubjectPriority

This table stores the priorities that are assigned to the user’s subjects.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary Key | Foreign Key | Field Name | Types | Description |
| SubjPriorityID | None | SubjPriorityID  SubjPriority | AutoNumber  Short Text | Unique identifier  Priority name |

### tblSubjectPriority Sample Data

|  |  |
| --- | --- |
| **SubjPriorityID** | **SubjPrioirty** |
| 1 | High |
| 2 | Medium |
| 3 | Low |

## tblSubject

This table stores the subjects that are assigned to the user’s tasks.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary Key | Foreign Key | Field Name | Types | Description |
| SubjectID | None | SubjectID  SubjectName  SubjPriorityID | AutoNumber  ShortText  Number | Unique identifier  Name of subject  ID of subject priority |

### tblSubject Sample Data

|  |  |  |
| --- | --- | --- |
| **SubjectID** | **SubjectName** | **SubjPriorityID** |
| 1 | CAT | 1 |
| 2 | IT | 2 |
| 3 | Art | 3 |

## tblTaskStatus

This table stores the status that is assigned to the user’s tasks.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary Key | Foreign Key | Field Name | Types | Description |
| TaskStatusID | None | TaskStatusID  TaskStatus | AutoNumber  ShortText | Unique identifier  Status name |

### tblTaskStatus Sample Data

|  |  |
| --- | --- |
| **TaskStatusID** | **TaskStatus** |
| 1 | Active |
| 2 | In Progress |
| 3 | Completed |

## tblUser

This table stores the user related information.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Primary Key | Foreign Key | Field Name | Types | Description |
| UserID | None | UserID  //UserCreateDate  UserEmail  UserFullName  UserPhone  UserPassword  //Active | AutoNumber  //ShortText  ShortText  ShortText  Number  Short Text  True/False | Unique identifier  //System timestamp  User email  User first, surname  User telephone nbr  User password  Tick box |

### tblUser Sample Data

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **UserID** | **UserCreateDate** | **UserEmail** | **UserFullName** | **UserPhone** | **UserPassword** | **//Active** |
| 1 | 01/04/2021 | [@outlook.com](mailto:ingesenekal@outlook.com) | Inge Senekal | 060 964 3852 | Password1 | Yes |
| 2 | 06/04/2021 | [@gmail.com](mailto:deborahsenekal@gmail.com) | Deborah Senekal | 079 495 7281 | Password2 | Yes |
| 3 | 08/04/2021 | [@cashkey.co.za](mailto:asenekal@cashkey.co.za) | Andre Senekal | 082 553 4427 | Password3 | Yes |



# Explanation of Storage Design

## Primary Storage

### User Class

This class is used to store the users’ information therefor it contains the username, password, email address and the phone number of the user. This information is extremely important since it if the user can not log into their account the program will not be able to run. It is possible for a user however to create a new account.

### Task Class

This class stores the most important information and is the reason for the program, it consist of all the details concerning the task such as the task name, task status and subject the task is in.

All information here will be used to create the final product in the planner and will aid the user on managing there.

## Secondary Storage

My whole program will be using a database.

A database is more appropriate since the data consist of complex structures and all data will be linked to different tables.

The database will consist of 6 tables.

tblUser

The users’ information will be used to log into the program therefor this table is the most important.

Users’ information will also be private therefor it can not be changed and only when a user registers will new data be entered into the table.

Since a user must be able to register the new data will have to be stored permanently for the user to be able to access their information again. Therefor a database is the most appropriate storage option. This data must be re-accessed if program is run again. An array must be created to hold information about the user.

tblTask

This table will be the link between other tables and therefor can be seen as the main table in the database.

Information in this table will be public therefor the user can alter the data.

The users task information consist of complex table designs and links must be made between tables therefor a database is appropriate.

Although since most of the information must be able to go through a search an array will have to be made for the tasks.

This table links between tblSubject, tblSubjectPriority, tblTaskStatus and tblUser

All these tables will be stored permanently in secondary memory.