High Level Design (HLD) Document

Kanban Board

The Kanban Board is a method to manage and improve work across human systems.

This approach aims to manage work by balancing demands with available capacity, and by improving the handling of system-level bottleneck.

Work items are visualized to give participants a view of progress and process, form start to finish via a Kanban Board.

Terminology:

Client – companies with up to 1,000 employees.

User – an employee with an authorization to the project given by the company.

Task/assignment – has the following fields:

1) The creation time.
2) A due date.
3) A title and a description.

Board – by default has three columns named – “Backlog”, “In Progress” and “Done”, but more columns can be added or removed.

User – identified by a unique email address and a login with a password.

Registration and login:

Registering new users must be possible at any time and existing users should have the option to login and logout.(functional)

When an employee is fired his user is automatically deleted.(non-functional)

Keeping a reasonable amount of users

An email is unique for the system and belongs to one user at most and in order to enter the system a user must put in the correct email and password.(functional)

Different identification and security

UX:

Limiting the maximum number of tasks should be supported by the system, however by default there will be no limit.(functional)

The user can manage a number of boards and select which one to present.(functional)

The tasks within a column can be sorted by due date or by creation date.(non-functional)

Tasks can be filtered by their text field(i.e. description and title).(non-functional)

Easy to use by the user

Task movement is only allowing between one column the next column (from the left).(functional)

Columns can be moved and switched.(functional)

All of the tasks data can be edited with the exception of the creation time unless the task reaches the last column.(functional)

A user can add and remove boards. (functional)

Keeping processes properly

Exceptions:

The program is expected to operate even when an error occurs : Handle any malformed input, handle logic error etc.(non-functional)

Avoid system crashes

Tests:

The program implements Unit Testing of it functions, in a fully covered manner. (non-functional)

Minimize possible malfunction

Security:

User will be disconnected automatically after 3 minutes of none action. (non-functional)

Helps keep the system safe

Persistency:

Persistence data (Users, tasks and columns) should be stored in a Data Base by serialization, and should be restored while the program starts. (non-functional)

Everything is restored if the system crashes

Logging:

Using a log that will track the errors in our system. An error is not just an exception, but also any action that is invalid in our domain. (non-functional)

improving the system and manage the users successfully

Presentation Layer

viewModel

ColumnDataContext

column:string
position:int
+PropertyChanged:PropertyChangedEventHandler
service:Service

+AddColumn():bool
+RemoveColumn():bool
+MoveColumn():bool

TaskIDWindowDataContext

taskID:string
+PropertyChanged:PropertyChangedEventHandler
service:Service

+PushTask():bool
+RemoveTask():bool
+ShowTask():Task
+ContainsTask():bool

LimitColumnWindowDataContext

nameOfColumn:string
tasks:int
+PropertyChanged:PropertyChangedEventHandler
service:Sercvice

+limitColumnTasks():bool

TaskWindowDataContext

title:string
descrition:string
date:string
+PropertyChanged:PropertyChangedEventHandler
service:Service

+addTask():bool

UserWindowDataContext

email:string
pwd:string
+PropertyChanged:PropertyChangedEventHandler
service:Service

+login():bool
+register():bool

ChangeTaskWindowDataContext

taskData:string
taskID:string
+PropertyChanged:PropertyChangedEventHandler
service:Service

+ChangeTitle():bool
+ChangeDescription():bool
+ChangeDueDate():bool

BoardWindowDataContext

-selectColumn:BoardWindowColumn
-selectRow:BoardWindowRow
-gridView:ICollectionView
-gridcolumn:ICollectionView
-columns:ObservableCollection<BoardWindowColumn>
-tasks:ObservableCollection<BoardWindowRow>
service:Service
+PropertyChanged:PropertyChangedEventHandler

-UpdateFilterTasks()
-UpdateFilterColumns()
+BoardWindowDataContext()
+showTheard()

BoardsDataContext

service:Service
-selectBoard:BoardsWindowColumn
-gridBoard:ICollectionView
-boards:ObservableCollection<BoardsWindowColumn>

+AddBoard():bool
+RemoveBoard():bool
+ShowBoard():bool
-UpdateFilterBoards()
+showTheard1()

BoardWindowRow

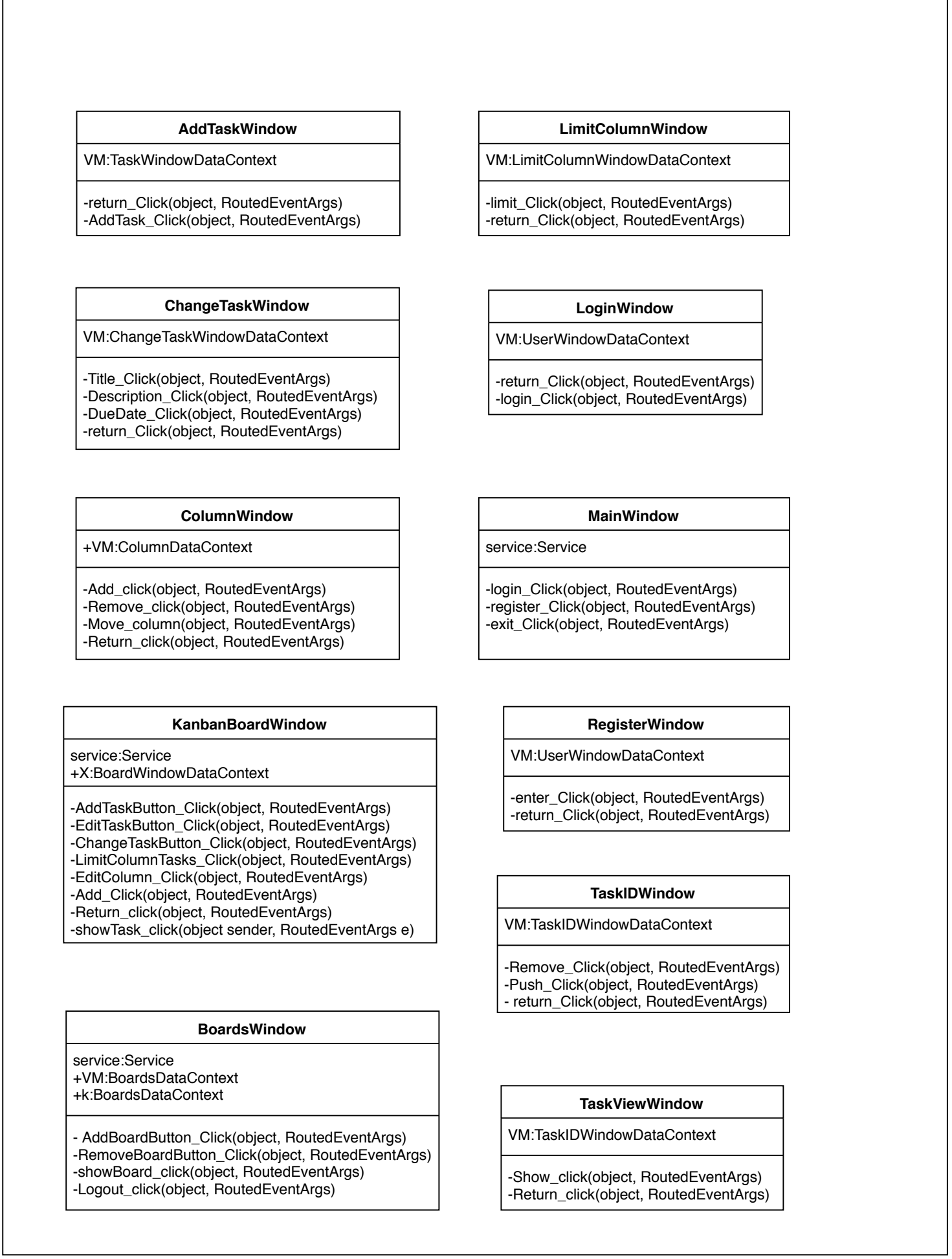
+NameOfColumn:string
+TaskID:string
+CreationDate:string
+DueDate:string
+Title:string
+Description:string

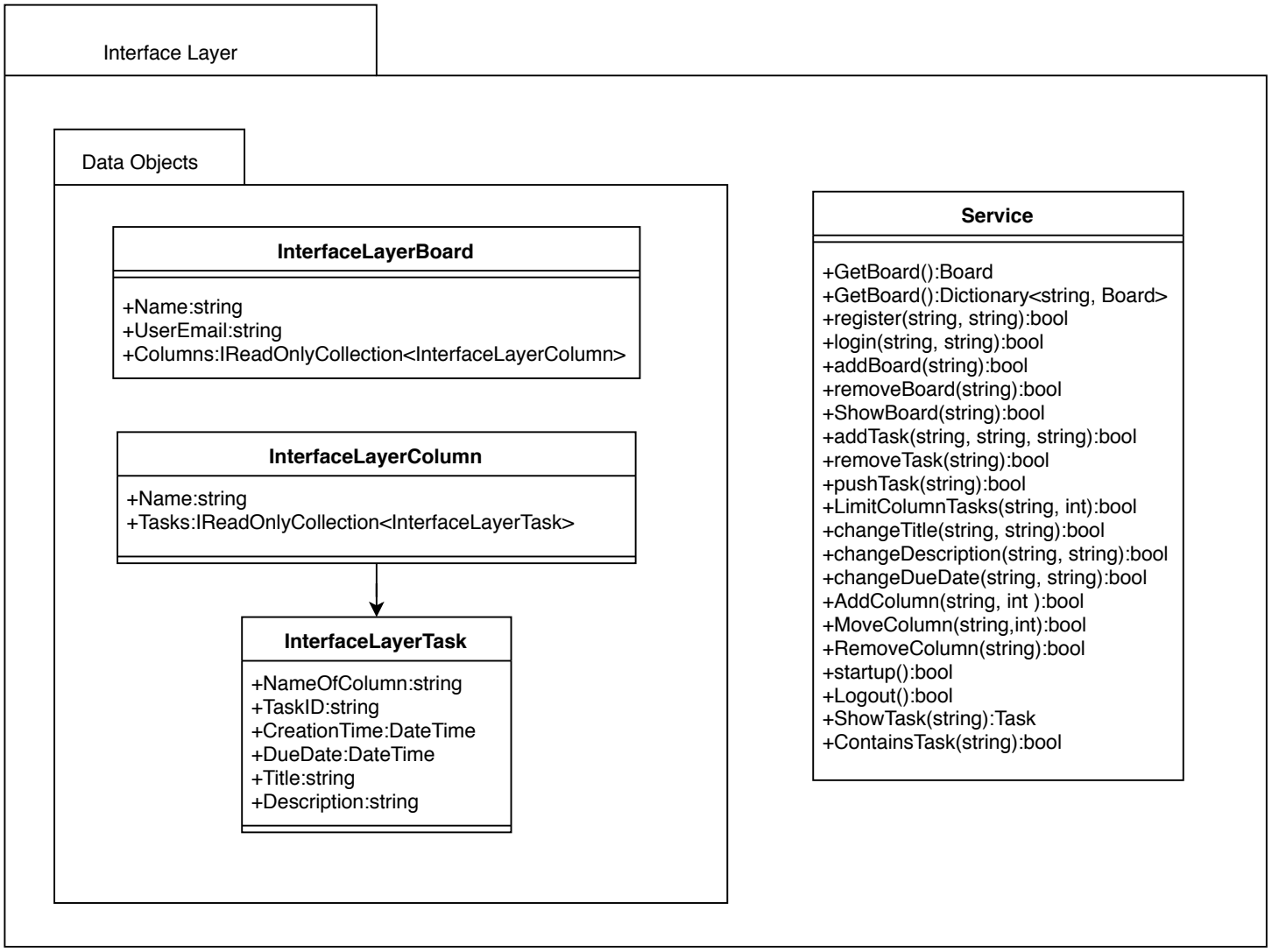
BoardWindowColumn

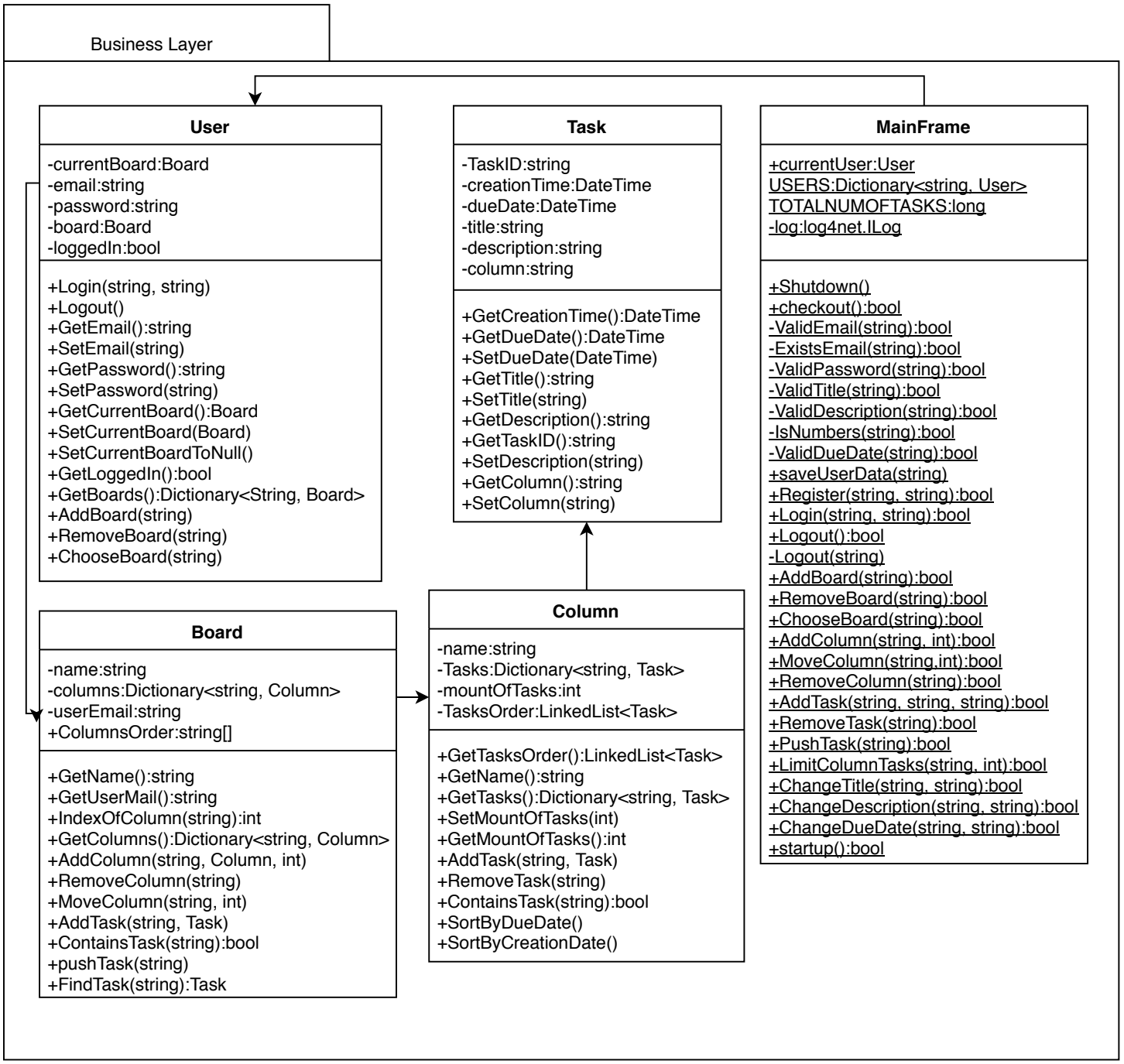
+ColumnName:string
+NumOfTasks:int
+LimitTasks:int

BoardsWindowColumn

+BoardName:string
+NumOfColumns:int


AddTaskWindow
VM:TaskWindowDataContext
-return_Click(object, RoutedEventArgs)
-AddTask_Click(object, RoutedEventArgs)
LimitColumnWindow
VM:LimitColumnWindowDataContext
-limit_Click(object, RoutedEventArgs)
-return_Click(object, RoutedEventArgs)
ChangeTaskWindow
VM:ChangeTaskWindowDataContext
-Title_Click(object, RoutedEventArgs)
-Description_Click(object, RoutedEventArgs)
-DueDate_Click(object, RoutedEventArgs)
-return_Click(object, RoutedEventArgs)
LoginWindow
VM:UserWindowDataContext
-return_Click(object, RoutedEventArgs)
-login_Click(object, RoutedEventArgs)
ColumnWindow
+VM:ColumnDataContext
-Add_click(object, RoutedEventArgs)
-Remove_click(object, RoutedEventArgs)
-Move_column(object, RoutedEventArgs)
-Return_click(object, RoutedEventArgs)
MainWindow
service:Service
-login_Click(object, RoutedEventArgs)
-register_Click(object, RoutedEventArgs)
-exit_Click(object, RoutedEventArgs)
KanbanBoardWindow
service:Service
+X:BoardWindowDataContext
-AddTaskButton_Click(object, RoutedEventArgs)
-EditTaskButton_Click(object, RoutedEventArgs)
-ChangeTaskButton_Click(object, RoutedEventArgs)
-LimitColumnTasks_Click(object, RoutedEventArgs)
-EditColumn_Click(object, RoutedEventArgs)
-Add_Click(object, RoutedEventArgs)
-Return_click(object, RoutedEventArgs)
-showTask_click(object sender, RoutedEventArgs e)
RegisterWindow
VM:UserWindowDataContext
-enter_Click(object, RoutedEventArgs)
-return_Click(object, RoutedEventArgs)
TaskIDWindow
VM:TaskIDWindowDataContext
-Remove_Click(object, RoutedEventArgs)
-Push_Click(object, RoutedEventArgs)
- return_Click(object, RoutedEventArgs)
BoardsWindow
service:Service
+VM:BoardsDataContext
+k:BoardsDataContext
- AddBoardButton_Click(object, RoutedEventArgs)
-RemoveBoardButton_Click(object, RoutedEventArgs)
-showBoard_click(object, RoutedEventArgs)
-Logout_click(object, RoutedEventArgs)
TaskViewWindow
VM:TaskIDWindowDataContext
-Show_click(object, RoutedEventArgs)
-Return_click(object, RoutedEventArgs)

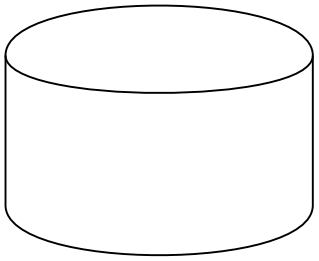

Interface Layer
Data Objects
InterfaceLayerBoard
+Name:string
+UserEmail:string
+Columns:IReadOnlyCollection<InterfaceLayerColumn>
InterfaceLayerColumn
+Name:string
+Tasks:IReadOnlyCollection<InterfaceLayerTask>
InterfaceLayerTask
+NameOfColumn:string
+TaskID:string
+CreationTime:DateTime
+DueDate:DateTime
+Title:string
+Description:string
Service
+GetBoard():Board
+GetBoard():Dictionary<string, Board>
+register(string, string):bool
+login(string, string):bool
+addBoard(string):bool
+removeBoard(string):bool
+ShowBoard(string):bool
+addTask(string, string, string):bool
+removeTask(string):bool
+pushTask(string):bool
+LimitColumnTasks(string, int):bool
+changeTitle(string, string):bool
+changeDescription(string, string):bool
+changeDueDate(string, string):bool
+AddColumn(string, int):bool
+MoveColumn(string,int):bool
+RemoveColumn(string):bool
+startup():bool
+Logout():bool
+ShowTask(string):Task
+ContainsTask(string):bool


Business Layer
User
-currentBoard:Board
-email:string
-password:string
-board:Board
-loggedIn:bool
+Login(string, string)
+Logout()
+GetEmail():string
+SetEmail(string)
+GetPassword():string
+SetPassword(string)
+GetCurrentBoard():Board
+SetCurrentBoard(Board)
+SetCurrentBoardToNull()
+GetLoggedIn():bool
+GetBoards():Dictionary<String, Board>
+AddBoard(string)
+RemoveBoard(string)
+ChooseBoard(string)
Task
-TaskID:string
-creationTime:DateTime
-dueDate:DateTime
-title:string
-description:string
-column:string
+GetCreationTime():DateTime
+GetDueDate():DateTime
+SetDueDate(DateTime)
+GetTitle():string
+SetTitle(string)
+GetDescription():string
+GetTaskID():string
+SetDescription(string)
+GetColumn():string
+SetColumn(string)
MainFrame
+currentUser:User
USERS:Dictionary<string, User>
TOTALNUMOFTASKS:long
-log:log4net.ILog
+Shutdown()
+checkout():bool
-ValidEmail(string):bool
-ExistsEmail(string):bool
-ValidPassword(string):bool
-ValidTitle(string):bool
-ValidDescription(string):bool
-IsNumbers(string):bool
-ValidDueDate(string):bool
+saveUserData(string)
+Register(string, string):bool
+Login(string, string):bool
+Logout():bool
-Logout(string)
+AddBoard(string):bool
+RemoveBoard(string):bool
+ChooseBoard(string):bool
+AddColumn(string, int):bool
+MoveColumn(string,int):bool
+RemoveColumn(string):bool
+AddTask(string, string, string):bool
+RemoveTask(string):bool
+PushTask(string):bool
+LimitColumnTasks(string, int):bool
+ChangeTitle(string, string):bool
+ChangeDescription(string, string):bool
+ChangeDueDate(string, string):bool
+startup():bool
Board
-name:string
-columns:Dictionary<string, Column>
-userEmail:string
+ColumnsOrder:string[]
+GetName():string
+GetUserMail():string
+IndexOfColumn(string):int
+GetColumns():Dictionary<string, Column>
+AddColumn(string, Column, int)
+RemoveColumn(string)
+MoveColumn(string, int)
+AddTask(string, Task)
+ContainsTask(string):bool
+pushTask(string)
+FindTask(string):Task
Column
-name:string
-Tasks:Dictionary<string, Task>
-mountOfTasks:int
-TasksOrder:LinkedList<Task>
+GetTasksOrder():LinkedList<Task>
+GetName():string
+GetTasks():Dictionary<string, Task>
+SetMountOfTasks(int)
+GetMountOfTasks():int
+AddTask(string, Task)
+RemoveTask(string)
+ContainsTask(string):bool
+SortByDueDate()
+SortByCreationDate()

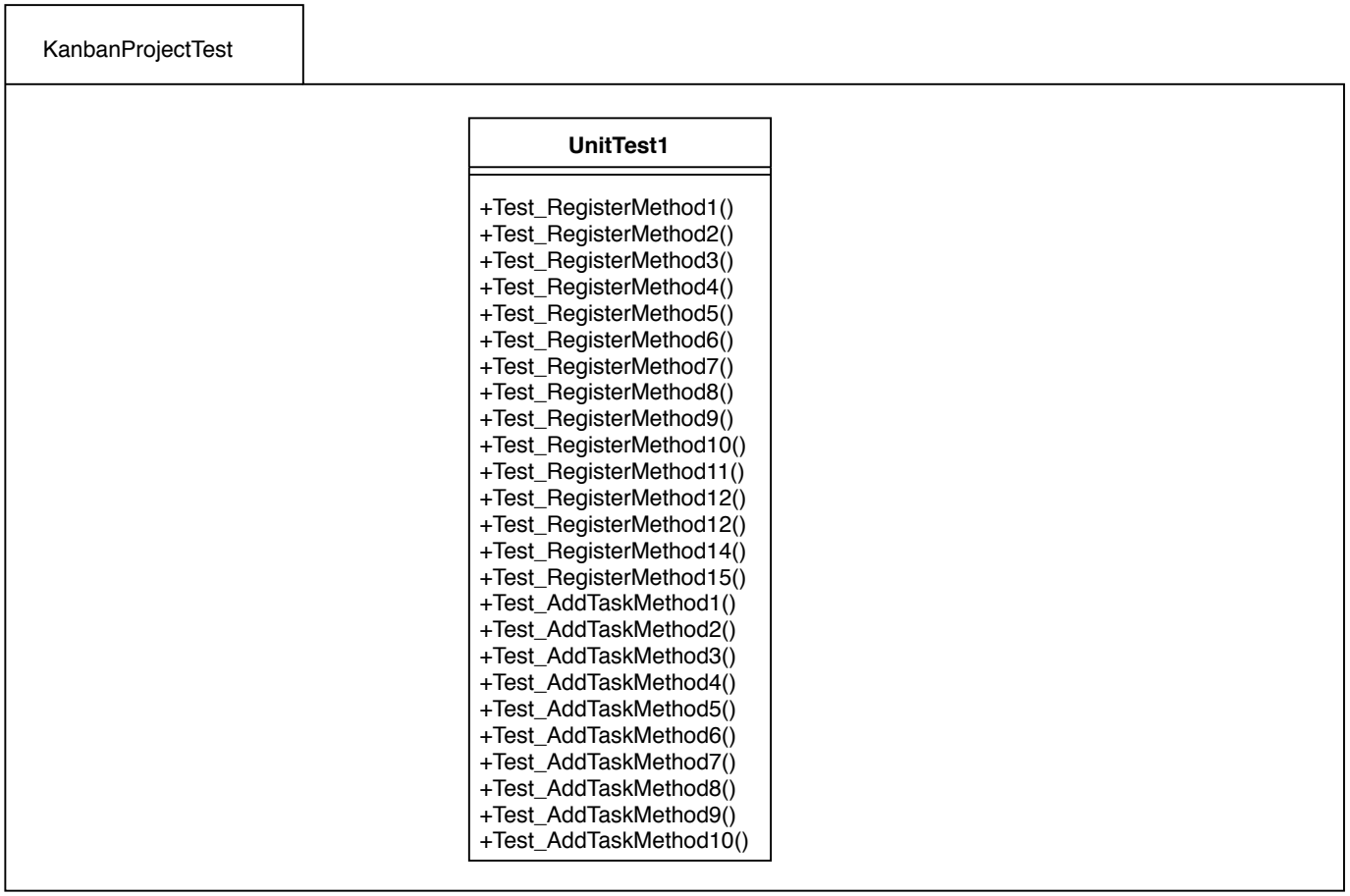
Data Layer

Serialization

+SaveTask(string,string,string,string,string,string,string,string)
+SaveUser(string,string)
+SaveBoard(string,string)
+SaveColumn(string,string,string,int,int)
+EditColumn(string,string,string)
-LoadUsers():Dictionary<string, User>
-LoadBoards():LinkedList<Board>
-mergeUserBoards():Dictionary<string, User>
+LoadData():Dictionary<string, User>
+RemoveTask(string)
+RemoveColumn(string,string,string)
+RemoveBoard(string,string)
-GetConnenctionString(string)



persistance storage


KanbanProjectTest
UnitTest1
+Test_RegisterMethod1()
+Test_RegisterMethod2()
+Test_RegisterMethod3()
+Test_RegisterMethod4()
+Test_RegisterMethod5()
+Test_RegisterMethod6()
+Test_RegisterMethod7()
+Test_RegisterMethod8()
+Test_RegisterMethod9()
+Test_RegisterMethod10()
+Test_RegisterMethod11()
+Test_RegisterMethod12()
+Test_RegisterMethod12()
+Test_RegisterMethod14()
+Test_RegisterMethod15()
+Test_AddTaskMethod1()
+Test_AddTaskMethod2()
+Test_AddTaskMethod3()
+Test_AddTaskMethod4()
+Test_AddTaskMethod5()
+Test_AddTaskMethod6()
+Test_AddTaskMethod7()
+Test_AddTaskMethod8()
+Test_AddTaskMethod9()
+Test_AddTaskMethod10()