University of Dhaka

Department of Computer Science and Engineering

Project Report

Object Oriented Programming Lab(CSE-2112)

Project Name

My App:Your Daily Tracker

Team Members:

1.
Name: Lubna Zahan Lamia
Roll:25
Department of Computer Science & Engineering
University of Dhaka.

2.
Name: Mabsur Fatin Bin Hossain
Roll:26
Department of Computer Science & Engineering
University of Dhaka.

3.
Name: Ahad Bin Islam Shoeb
Roll:27
Department of Computer Science & Engineering
University of Dhaka.

Project Coordinators

1.Dr Muhammad Ibrahim
Assistant Professor
Department of Computer Science and Engineering
University of Dhaka.

2.Md.Ashraful Islam
Lecturer
Department of Computer Science and Engineering
University of Dhaka

GitLink:https://github.com/mansif08/MyApp-Your-Daily-Assistant

Requirement Analysis:

In this age of science and technology, information and systems to keep track of our daily life,ease daily household and family management much more convenient than ever before.Family,being the main unit of a society,is a hub of information and data about the health,career,economic condition and correlation of its members. Efficient management of these information ensures quality life of the members of the family using technology.

Purpose:

Our app “My App-Your Daily Tracker” is an app that consists of different apps that helps a user to store information about his/her daily tasks and keep track of his/her health.Our app interrelates daily tasks management and health tracking system that emphasises on storing informations as well as producing suggestions evaluating the data provided by the user. Also ,our app has features that enables a user to track budget and carry out calculation related tasks.A child-parent relationship is maintained that gives the authority to a parent to add instructions for his/her child,manipulating the child’s daily tasks.My app therefore intends to ease family management through storing and interrelating information about a user as well as maintaining the parent-child relationship-which is a major part of family management.

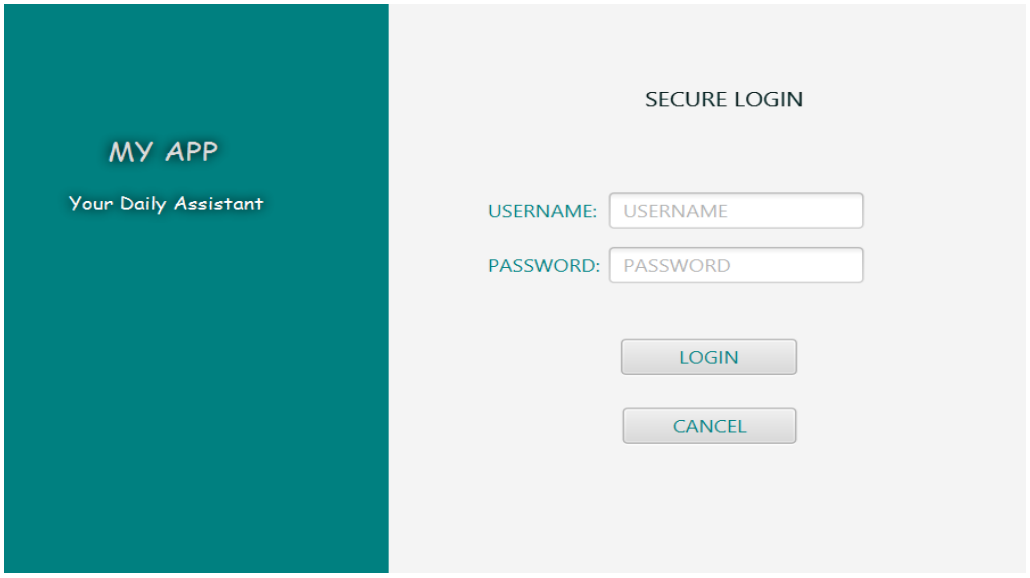
Intended Users:

Members of any single/joint family maintaining the parent-child relationship through the registration system that gives authority to a parent only to register another user as parent.No child can register a user as parent because parent are provided with extended features that enables them to manipulate child’s daily tasks though instructions.This “My app” is a family oriented app.

Options:

1.Login System:

A login window is opened that allows previously registered members to Login through their provided username/password during registration.
User can cancel the process of login clicking on the “Cancel” button.

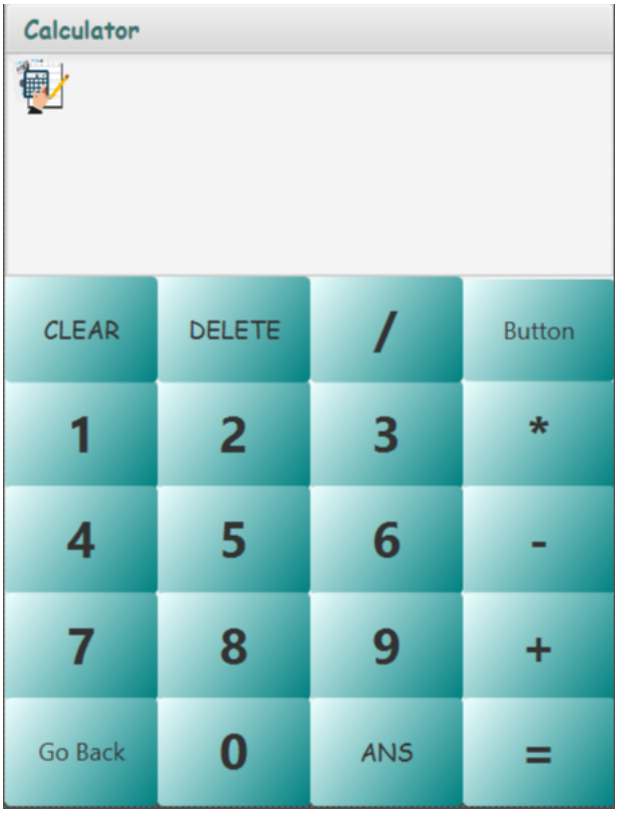


2.Main Menu:

Upon successfully logging in,a main menu appears that comprises different buttons to enter all of the sub apps-Calculator,To-do List,Health tracker,Budget Tracker and the Registration System.

3.Calculator:

A basic calculator that performs basic calculation related operations and also has features to erase one or more numbers and also features to clear out the entire display part of the calculator.

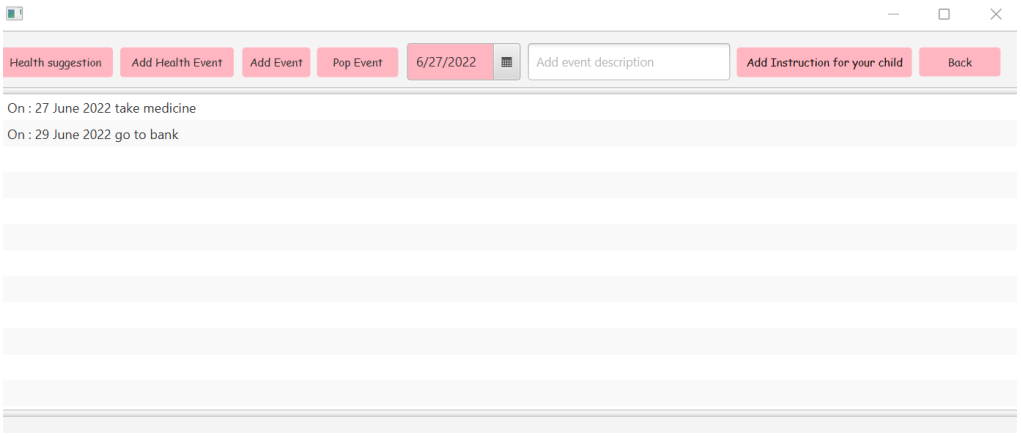


4.To-do list:

Reminder:

- ➢ A reminder page opens upon entering “To-Do List”
- ➢ “Go To To-Do List” button is provided to enter the main To-Do List upon clicking.

For Normal User/Child:

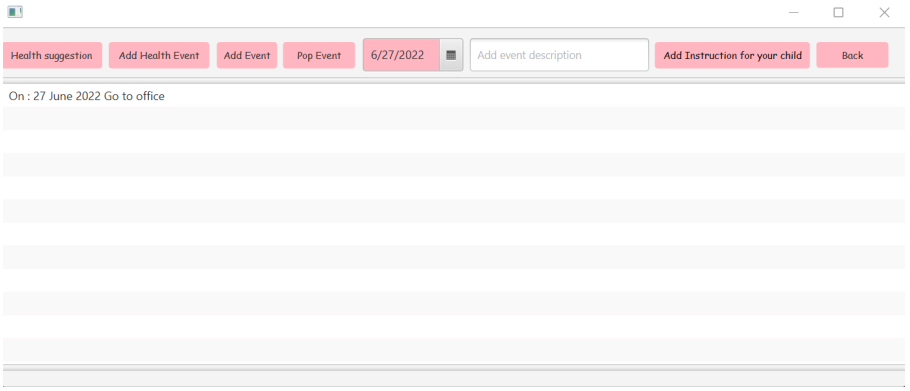


- Date Selection:
  Enables a user to select any date from the calendar to save tasks related to that date.
- Description:
  Enables a user to add a description of the task to be added in the To-do List.
- Add event:
  Option to add a task with the selected date and typed description.
- Pop event:
  Option to remove/delete a task from the list.
- Add Health Event:
  - ➢ Tasks related to health are saved in the list through this button.
  - ➢ Upon completion when a health event is popped out from the list,the task is saved in “Health Tracker”s “ Taken for Health Care Steps” section as a task that was completed for health related matters.

- Health Suggestions:
  A new window is opened if this button is clicked,this button creates connection with “Health Tracker” as health suggestions are shown in the window and suggestions are produced evaluating the health parameters that are either calculated(BMI) or provided(Pulse/Blood Pressure) of a user.

For Parent:

Along with the features described above,an additional feature is added described below -



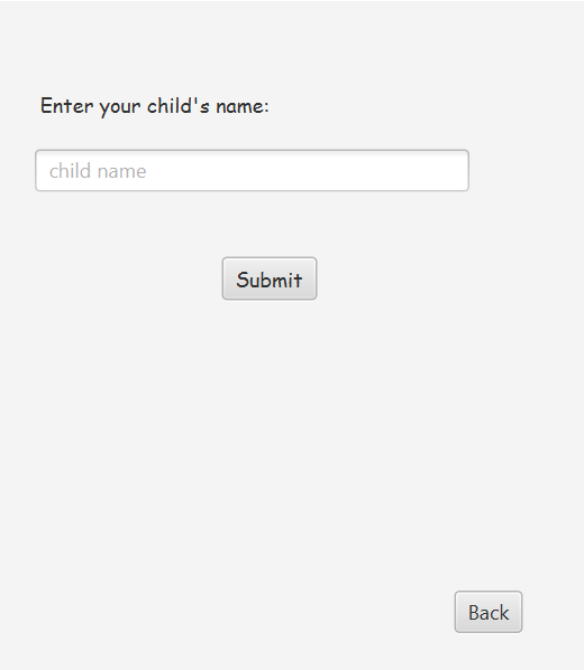
Here,a “Not authenticated to give instructions as a parent” message is shown if a child clicks “Add Instructions for your child” button is clicked.

- Add Instruction for Your Child:
  This enables a parent to add tasks/instructions to a child's to-do list.The child can see instructions anytime upon logging in.

  Enter your child's name:

➢ Allows parents to give instructions in the "To-do List" of a particular child.

➢ Upon entering the child's name,To-do List of that specific child is opened enabling the parent to add instructions in the child's to-do list.



5.Health Tracker:

- Calculate your BMI:

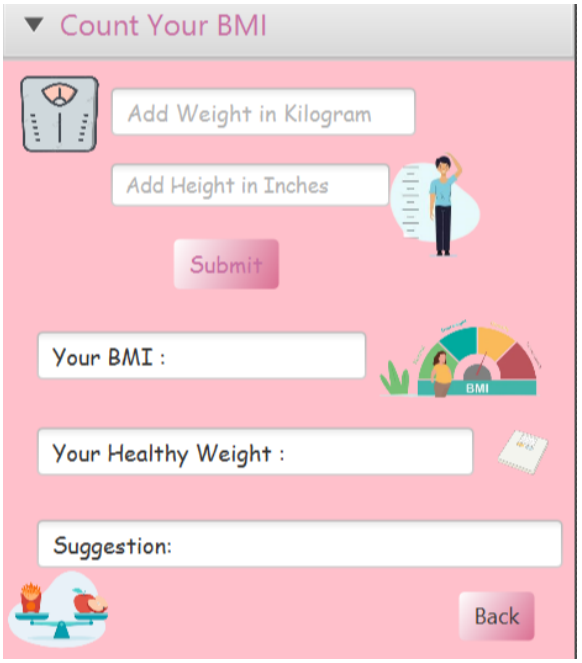
➢ Upon opening this subsection of the " Health Tracker"-options to enter weight and height is shown and upon submitting these datas,BMI of the user is calculated and shown

➢ .Also,the desired/moderate weight of the user is calculated and shown.

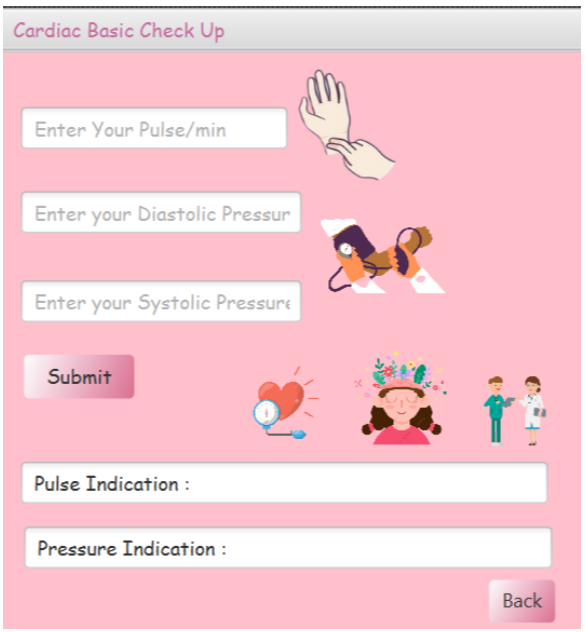
➢ Suggestions evaluating the BMI are shown in the "Suggestions" sections.

➢ Suggestions evaluating the BMI are also saved in the “My Health Suggestions” section of “To-Do-List”.

➢ It is to be noted that,if any invalid input(Negative value) is provided by the user,a warning message is shown indicating the value is invalid.



- Basic Cardiac Check up:
  - ➢ Pulse ,Systolic and diastolic blood pressure are taken as inputs,upon submitting these datas,the condition of pulse and pressure is shown in the corresponding section.
  - ➢ Also,evaluating the pulse and pressure provided by the user,suggestions to maintain these are shown in the “My Health Suggestions” section of “To-Do-List” if any abnormality in provided pulse rate and pressure is found.
  - ➢ It is to be noted that,if any invalid input(Negative value) is provided by
  - ➢ the user,a warning message is shown indicating the value is invalid.



- Exercises for you:
  Upon clicking this button,a new window appears which includes the following buttons-

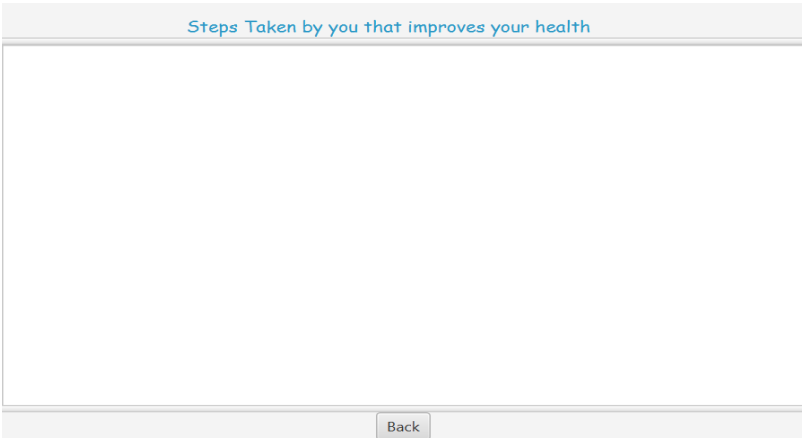
➢ Aerobic
➢ Strength Building
➢ Endurance
➢ Flexibility
➢ Cardio

Upon clicking one of these,images containing the instructions to carry out exercise of that specific type is shown.

- My Health Info:
  Information of health parameters provided(Pulse/Systolic Pressure/Diastolic Pressure) or calculated (BMI) is saved and upon clicking ,is shown in this section .

- Taken Health Care Steps:

  When a health event is added and completed in a “To-Do-List’-that event is stored and upon clicking,shown in this section of the health tracker.



6.Interrelation between “To-Do List” and “Health Tracker”:

➢ Related health suggestions are shown in the “Health Suggestion” part of the “To-Do List”,evaluating the health parameters(BMI,Pulse rate,Systolic/diastolic pressure) calculated/provided by the “Health Tracker”.

➢ If any event is added in the To-Do List using the “Add Health Event” button and is completed,it will be printed as “Steps taken for health” section in “Health Tracker”.

7.Interrelation between “Parent To-Do List” and “Child To-Do List”:

- ➢ A parent can enter instructions and get access to a child's To-Do List upon clicking the “Add Instructions for Child” button.
- ➢ Note that,a window appears upon clicking this button where the parent has to input correct child username.

8.Interrelation between components of “Health Tracker”:

- ➢ Data entered in the “Basic Cardiac Check up” is saved in “My Health Info” part of the health tracker along with date and time.
- ➢ Evaluated BMI in the “Calculate your BMI” is saved in “My Health Info” part of the health tracker along with date and time

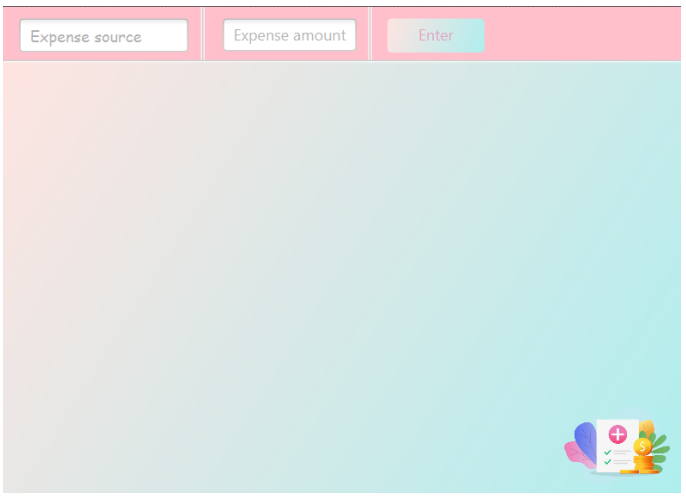
9.Budget Tracker:

- Track Income:

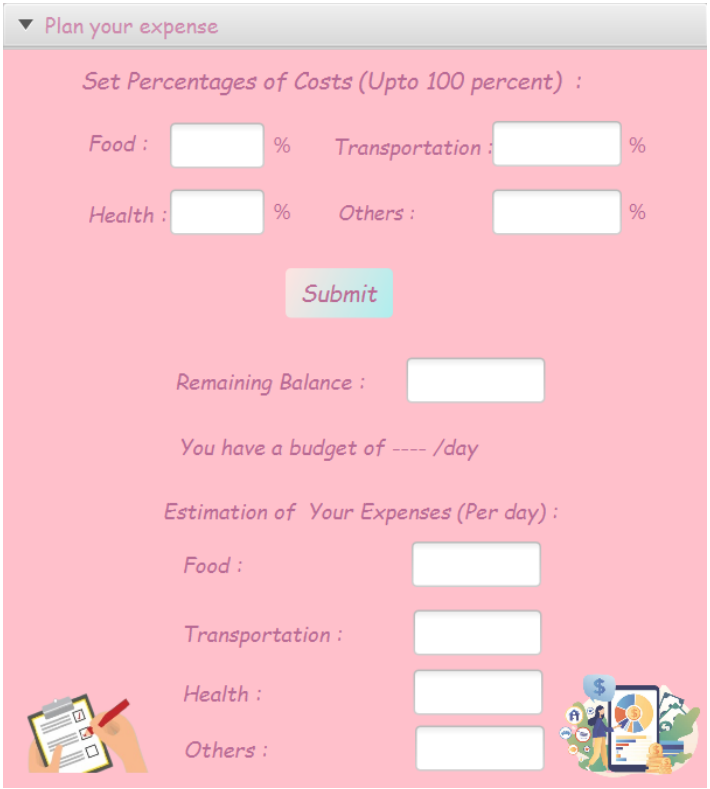
  This subsection takes income amount and income source as inputs and saves it as a list as well as shows the total income adding the new income input with the previous ones.

- Track Expense:

  This subsection takes expense amount and expense source as inputs and saves it as a list as well as shows the total expense adding the new expense input with the previous ones.

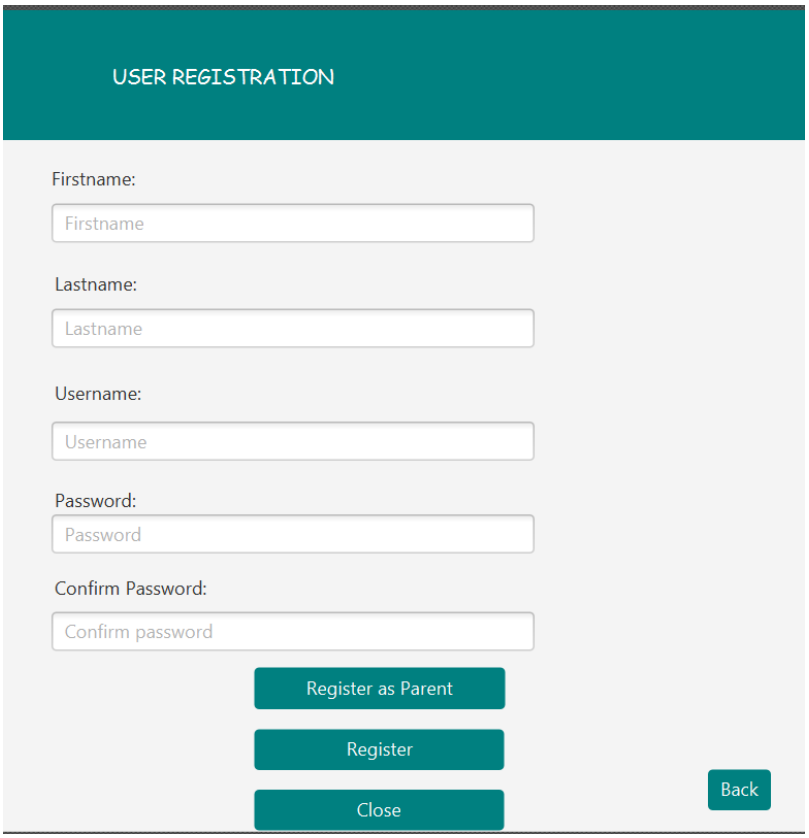


- Plan Your Expenses:
  - ➢ A user can input the percentage he/she wants to put of his total balance in a specific field(Food,transportation,health and others).
  - ➢ Feature to show the remaining balance is shown.
  - ➢ Feature to estimate a budget per day is shown.
  - ➢ Estimated amount that a user can use per day is shown in the specified sections.

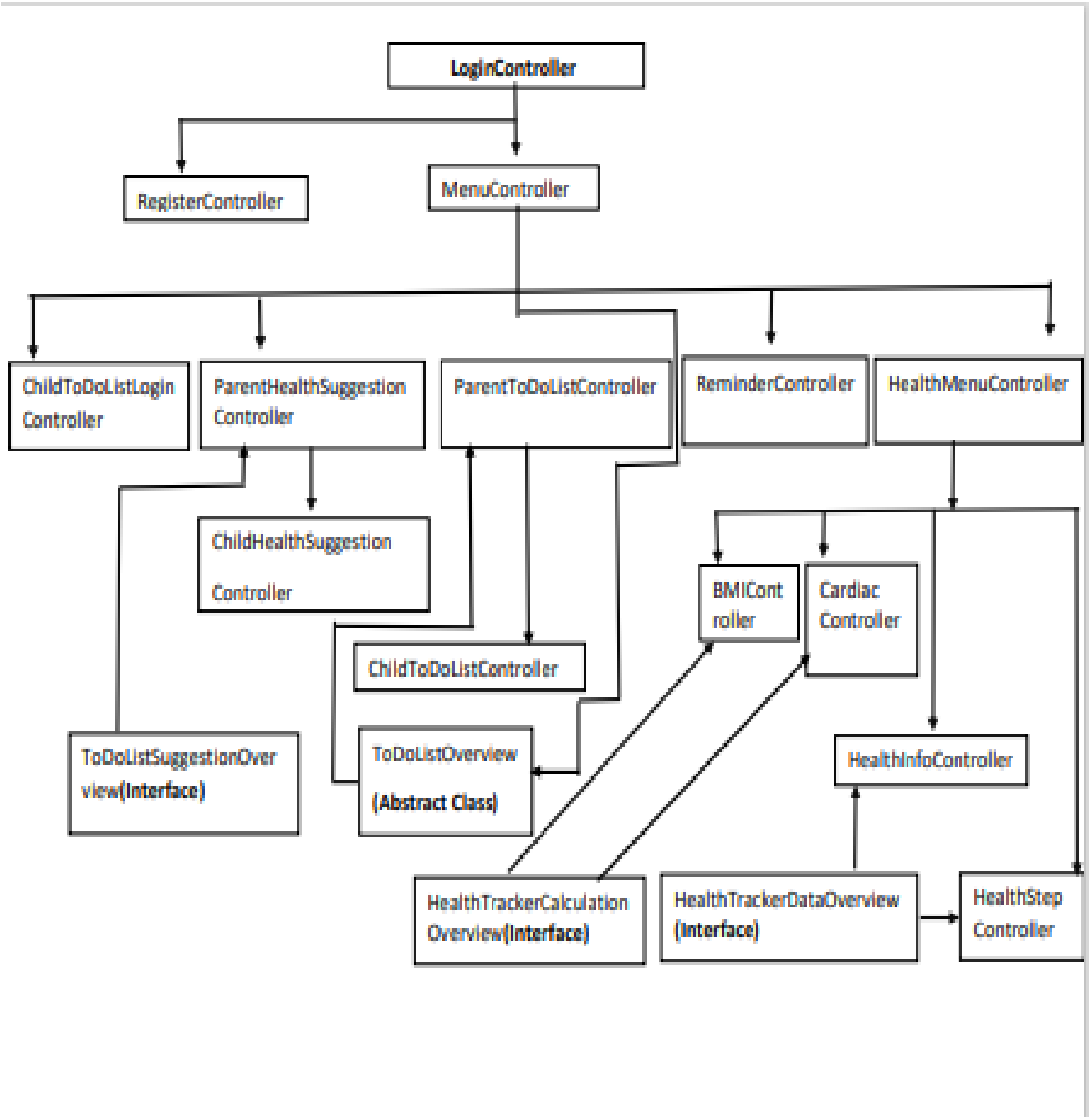


10.Register:

- A registration system to register in “My App”.
- Firstname,Lastname,Username,Password is taken as inputs.
- Confirm password section is added to retype the password .If passwords match,a user can be registered upon clicking the register button.
- The registration system that authority to a parent only to register another user as parent.If a child tries to register as parent “Not authenticated to register as parent” message is shown.



SYSTEM DESIGN:


LoginController
RegisterController
MenuController
ChildToDoListLogin Controller
ParentHealthSuggestion Controller
ParentToDoListController
ReminderController
HealthMenuController
ChildHealthSuggestion Controller
BMICont roller
Cardiac Controller
ChildToDoListController
ToDoListSuggestionOver view(Interface)
ToDoListOverview (Abstract Class)
HealthInfoController
HealthTrackerCalculation Overview(Interface)
HealthTrackerDataOverview (Interface)
HealthStep Controller

LoginController:

Methods:

LoginActionEvent():

- ➢ In this method,it is checked whether “Username” and “Password” is blank or not.
- ➢ If not, “ValidateLogin” method is called.

ValidateLogin():

- ➢ In this method,it is checked that whether the provided “Username” and “Password” is registered in the database or not.
- ➢ If the user enters valid data, then “Main menu” is opened.Otherwise,Invalid login warning is shown.

EnterMenu():

- ➢ It is the method that implements enter to main menu.

RegisterController:

Methods:

RegistrationButtonAction():

In this method,it is seen whether the information provided by the user is valid or not of the registration form.It checks whether any field left empty and whether the password in the password reconfirmation box matches with the first one or not.If every data provided by the user is valid,the user will be eligible for registration and a method to register the user will be called.

RegistrationParentButtonAction():

It does the same function as RegisterButtonAction but this method can be implemented only by parent users upon clicking the corresponding “Register as Parent” button,otherwise,if clicked by a normal user,a message is shown.The user who is being registered will be eligible for registering as a parent if being registered by another parent.

registerUser():

The data provided in the registration form and upon being validated by RegistrationButtonAction method,a user’s data will be registered into the database.

registerParentUser():
The data provided in the registration form and upon being validated by RegistrationParentButtonAction method,a user's data will be registered into the database as a parent.

MenuController:

Methods:

GoToCalculator():
Goes to the calculator section.

GoToToDoList():
Goes to the To-Do List section.

GoToHealthTracker():
Goes to the Health Tracker section.

GoToBudgetTracker():
Goes to the Budget Tracker section.

GoToRegister():
Goes to the Register section.

ChildToDoListLoginController:

Methods:

VerifyChildLogin():
Checks whether the name entered by the user is a child user's username or not.If eligible,child's To-Do List is opened.

ToDoListOverview(Abstract class):
As this is an abstract class,the methods are not implemented here.

Methods:

abstract public void fetchData()
abstract public void addEvent()
abstract public void addHealthEvent()

abstract public void popEvent()
abstract public void popEvent()
abstract public void refresh()
abstract public void addEventToDatabase()
abstract public void addEventToDatabase(String s)
abstract public void ToDoGoBack()
abstract public void GoToHealthSuggestion()
abstract public void GoToChildInstruction()

ParentToDoListController:

Methods:

Methods in this class are overridden from methods of ToDoListOverview class's method:

fetchData() :
Take unfinished tasks from the database and show it in the database.
addEvent() :
Adds tasks/events in the To-Do List.
addHealthEvent() :
Adds health event in the To-Do List.
popEvent() :
Pops out an event from the To-Do List.
addEventToDatabase() :
Adds the event/task to the database.
addEventToDatabase(String ItIsHealthEvent) :
Adds the event/task to the database.
GoToChildInstruction() :
Open child login page.
GoToHealthSuggestion() :
Opens health suggestions page where suggestions are shown according to health tracker's health parameters for the users to add health events accordingly.

ChildToDoListController:

Methods:

addEvent():
Same as super class's(ParentToDoListController) method.
addHealthEvent():

Same as super class's(ParentToDoListController) method.

popEvent():

Same as super class's(ParentToDoListController) method.

addEventToDatabase():

Same as super class's(ParentToDoListController) method.Saves the event in child database.

addEventToDatabase(String ItIsHealthEvent):

Same as super class's(ParentToDoListController) method.Saves the health event in child database.

fetchData():

Same as super class's(ParentToDoListController) method.Reads data/event from child database.

GoToHealthSuggestion() :

Same as super class's(ParentToDoListController) method.Shows suggestions according to child's health parameters from the health tracker.

ToDoListHealthSuggestionOverview(Abstract class):

As this is an abstract class,the methods are not implemented here.

Methods:

void HealthSuggestionGoBack();
void fetchData();

ParentHealthSuggestionController:

Methods:

fetchData():

Fetches health information from database and prints health suggestions accordingly (parent user)

ChildHealthSuggestionController:

Methods:

fetchData():

Fetches health information from database and prints health suggestions accordingly(child user)

ReminderController:

Methods:

MoveToToDoList():

Prints unfinished events/tasks in To-Do List.

fetchData():

Shows unfinished data of the corresponding date in the To-Do List.

HealthMenuController:

Methods:

GoToBMI():

Opens BMI window.

GoToCardiac():

Opens cardiac window.

GoToExercise():

Opens exercise window.

GoToHealthInfo():

Opens health info window.

GoToHealthSteps():

Opens steps taken for health window.

HealthTrackerCalculationOverview(Interface):

As this is an interface,the methods are not implemented here.

Methods:

void HealthTrackerGoBack()
void AddHealthReportToDatabase()
default void CalculateBMI()
default void CalculatePulseAndPressure()

BMIController:

Methods:

CalculateBMI():
Takes weight and height as inputs,calculate BMI and prints BMI suggestions accordingly.

AddHealthReportToDatabase():
Saves the calculated BMI in the database along with the time the data was read.

CardiacController:

Methods:

CalculatePulseAndPressure():
Takes pulse and systolic and diastolic pressure as inputs and prints pulse and pressure indications accordingly.

AddHealthReportToDatabase():
Saves the pulse and pressure in the database along with the time the data was read.

HealthTrackerDataOverview(Interface):

As this is an interface,the methods are not implemented here.

Methods:

void HealthTrackerGoBack();
void fetchData();

HealthInfoController:

Methods:

fetchData():
Health data is shown along with date measured on the “My Health Info” window according to username.

HealthStepController:

Methods:

fetchData():
Health step data is shown along by checking from the database whether an event related to health from To Do List is completed or not.

CalculatorController:

Methods:

deleteLst():
Delete the last number entered.
clearExp():
Clears the calculator screen.
getExp():
Returns the entered mathematical expression.
getResult():
Returns the result.
setResult():
Result expression is set here.
insertAnswer():
Expression is set here.
OnMouseClick(MouseEvent mouseEvent):
Identification of any button clicked on the calculator and relevant function is called here.

BudgetController:

Methods:

handleButtonAction:
Income tracker is opened upon clicking the “Track Income”button from the Budget Tracker menu.

handleButtonAction2:
Expense tracker is opened upon clicking the “TrackExpense” button from the Budget Tracker menu.

handleButtonAction3:
Planner is opened upon clicking the “Plan Your Expenses”button from the Budget Tracker menu.

Discussion:

Access Control:

In loginController,username and Childusername of the user are declared as private. But the getter and setter methods of username and Childusername are declared as public in loginController.

Setter method of username is used in
1.validateLogin() of LoginController

Getter method of username is used in
1.fetchData() of ParentHealthSuggestionController class.
2.fetchData(),popEvent(),addToDatabase(),addToDatabase(String s) goToChildInstruction() of ParentToDoListController.
3.fetchData() of ReminderController.
4.AddHealthReportToDatabase() of BMIController
5.AddHealthReportToDatabase() of CardiacController
6.fetchData() of HealthInfoController
7.fetchData() of HealthStepController

Setter method of Childusername is used in
1.VerifyChildLogin() of ChildToDoListLoginController

Getter method of Childuser is used in
1.VerifyChildLogin() of ChildToDoListLoginController
2.fetchData() of ChildHealthSuggestionController.
3.fetchData(),popEvent(),addToDatabase(),addToDatabase(String s) goToChildInstruction() of ChildToDoListController.
.
In BMIController, weight and height have been declared private.
In CardiacController, pulse ,diastolic pressure and systolic pressure have been declared private.

Inheritance:

loginController:
loginController contains the username and Childusername.So,every class in the Hierarchy extends loginController and its subclasses.
Extended by
1.registerController
2.menuController.

MenuController:
Extended by
1.ChildToDoListController
2.ParentHealthSuggestionController
3.ReminderController.
4.HealthMenuController.

ToDoListHealthSuggestionOverview(Interface):

Implemented by-
ParentHealthSuggestionController.

ParentHealthSuggestionController:
Extended by-
ChildToDoListController

ToDoListOverview(abstract class):
Extended by-
ParentToDoListController

ParentToDoListController:
Extended by-
ChildToDoListController
(Here,addEvent and addHealthEvent are not implemented in ChildToDoListController.The version of parent class of these function are called here.)

HealthMenuContoller:
Extended by
1.BMIController
2.CardiacController
3.HealthInfoController
4.HealthStepController

HealthTrackerCalculationOverview(interface):
Extended by
1.BMIController
2.CardiacController

HealthTrackerDataOverview(interface):
Extended by
1.HealthInfoController
2.HealthStepController

In BudgetTracker package:
1.localEvent2 is extends localEvent.
2.PlanController is extends localEvent2.

Polymorphism:

Overloading:
In ParentToDolistController and ChildToDoListController,
addEventToDatabase() and addEventToDatabase(String s) is overloaded.

Overriding:
1.In ParentToDoListController,
fetchdata(),addEvent,addHealthEvent(),popEvent(),refresh(),
addEventToDatabase(),addEventToDatabase(String s),
GoToChildInstruction() is overridden.

2.In ChildToDoListContoller,
fetchdata(),addEvent,addHealthEvent(),popEvent(),refresh(),
addEventToDatabase(),addEventToDatabase(String s) is overridden.

3.In ParentHealthSuggestionController,
fetchData() and HealthSuggestionGoBack() is overridden.

4.In ChildHealthSuggestionController,
fetchData() and HealthSuggestionGoBack() is overridden.

5.In BMIController,
HealthTrackerGoBack(),AddHealthReportToDatabase(),
CalculateBMI() is overridden.

6.In CardiacController,
HealthTrackerGoBack(),AddHealthReportToDatabase(),
CalculatePulseAndPressure() is implemented.

7.In HealthInfoController,
fetchData() and HealthTrackerGoBack() is implemented.

8.In HealthStepController,
fetchData() and HealthTrackerGoBack() is implemented.

9.In LocalEvent,LocalEvent2 and HealthEvent
toString() is overridden.

Interface and Abstract class:

ToDoListOverview(Abstract Class) :

This class gives an overview about the functionalities of to-do-list.
Abstract methods are-
abstract public void fetchData()
abstract public void addEvent()
abstract public void addHealthEvent()
abstract public void popEvent()
abstract public void popEvent()
abstract public void refresh()
abstract public void addEventToDatabase()
abstract public void addEventToDatabase(String s)
abstract public void ToDoGoBack()
abstract public void GoToHealthSuggestion()

Now another method is-
public void GoToChildInstruction()

This method has an implementation as it is only implemented by ParentToDoListController.As ChildToDoList does not implement it, It is not abstract.

ToDoListHealthSuggestionOverview(Interface):

This interface gives the overview of the health suggestion section given on Parent and child to-do-list.

Methods are:
void HealthSuggestionGoBack();
void fetchData();

Number of constants are declared which will be used in calculations.
The constants are of:
1.IdealBMI range
2. Underweight, Overweight,ObesityClass1 and ObesityClass2 range.
3.Normal Pulse range.
4.High pressure and low pressure range.

HealthTrackerCalculationOverview(Interface):
It gives an overview of the calculation related to Health Tracker.

Methods are:
void HealthTrackerGoBack()
void AddHealthReportToDatabase()
default void CalculateBMI()
default void CalculatePulseAndPressure()

Number of constants are declared which will be used in calculations.
The constants are of:
1.IdealBMI range
2. Underweight, Overweight,ObesityClass1 and ObesityClass2 range.
3.Normal Pulse range.
4.Normal pressure range.
5.Inch Per Meter.

HealthTrackerDataOverview(Interface):
It gives overview of the data shown related to Health Tracker to the user..

Methods are:
void HealthTrackerGoBack();
void fetchData();

Exception Handling:

1.Exception when loading FXML file is handled in every controller.

2.Exception in connecting with database system is handled in loginController,RegisterController,ChildToDoListLoginController, ParentHealthSuggestionController,ChildHealthSuggestionController, ParentToDoListController,ChildToDoListController,ReminderController, BMIController,CardiacController,HealthStepController,HealthInfoController.

3. In loginController, loginActionEvent checks if the username and password are empty and generate warning message .validateLogin gives warning message if the Login was invalid.

4. In registerController,registrationButtonAction generates warning message if firstname,lastname,username and setpassword are empty as well as matches the two given password by the user.
registrationParentButtonAction does the same activity and checks if the user is a parent who is trying to register a new parent.

5. In ParentToDoListController, addEvent and addHealthEvent check if the task added is empty.

6. In ParentToDoListController and ChildToDoListController ,popEvent checks If the list is empty when invoking pop event .

7. In BMIController, CalculateBMI throws runtime exception if height and weight is not positive.

8. In CardiacController,CalculatePulseAndPressure throws runtime exception if pulse,diastolic pressure and systolic pressure is not positive.

9. In EvaluateString of DailyCalculator package , applyOp throws runtime Exception if a number is divided by zero.

10. In LocalEvent of BudgetTracker package, LocalEvent(String ,String) constructor throws runtime exception if income is given non positive.

11. In LocalEvent2 of BudgetTracker package,LocalEvent2(String , String) constructor throws runtime exception if expense is given non positive.

12. In PlanController of BudgetTracker package, HandleSubmit throws runtime exception if food percentage ,transport percentage,health percentage,others percentage are non positive and the sum of these percentages is not 100.

13. In PlanController of BudgetTracker package, HandleSubmit throws Arithmetic exception if the total is non positive.

File I/O:

File I/O related tasks are covered by MYSQL database.

String Operations:

Methods of String class like are used throughout the project.
1. char charAt(int index)
2. int length()
3. boolean equals()
4. boolean isEmpty()
5. String concat(String str)
6. String valueOf(double d)
7. String valueOf(int d)
8. DateFormat.format() is used to convert date to string

Javafx collection:

Observable list is used in ChildToDoListLoginController, ParentHealthSuggestionController,ChildHealthSuggestionController, ParentToDoListController,ChildToDoListController,ReminderController, HealthStepController,HealthInfoController.

Static Keyword in java:

In loginController,userName and childUsername have been declared static. So,every object of loginController can share the same userName and childUsername.

Total income and total expense of localEvent and localEvent2 respectively have been made static so that values are saved while going to other windows.

Multiple inheritance of interfaces:

In java,multiple inheritance of classes except interfaces are not allowed. ParentHealthSuggestionController implements both Initializable and ToDoListHealthSuggestionOverview.

ParentHealthSuggestionController implements both Initializable and ToDoListHealthSuggestionOverview.

HealthInfoController and HealthStepContoller implements both Initializable and HealthTrackerDataOverview.

Conclusion and future work:

With the limited time and resources we had,we tried to ‘My App’ a family oriented app that incorporates efficient management of information and ensures quality life of the members of the family using technology.In future we plan to develop more features and make the app accessible to a broader audience.The extended features we plan to incorporate are-

1.Extend the information exchange and interrelation between users-for example,different family members such as spouses,grandparents,grandchildren etc can add data(example-Instruction) in one anothers To-Do-Lists which we have done in case of parents and children only.

2.Though we have made my app for family use right now,we plan to extend its use in other fields such as -educational field(Maintains teacher-student,staff-authority information and correlation between the),work field(Instruction chain among higher authority and staffs) etc.

3.Adding mental health section.

4.Making accessible to a broad audience and taking feedback from them.

The End