Module Interface Specification for REVITALIZE

Team 13, REVITALIZE
Bill Nguyen
Syed Bokhari
Hasan Kibria
Youssef Dahab
Logan Brown
Mahmoud Anklis

January 17, 2023

1 Revision History

Date	**Version**	**Notes**
Date 1	1.0	Notes
Date 2	1.1	Notes

2 Symbols, Abbreviations and Acronyms

See SRS Documentation at https://github.com/BillNguyen1999/REVITALIZE/tree/main/docs/SRS

symbol	description
REVITALIZE	Name of application
UAT	User Acceptance Testing
UI/UX	User Interface/User Experience
HCI	Human-Computer Interface
MG	Module Guide
MIS	Module Interface Specification
SRS	Software Requirements Specification
VnV	Verification and Validation
LP	Login Page
SP	Sign-up Page
MP	Main Page
DS	Diet Section
WS	Workout Section
RS	Rest Section

Contents

3 Introduction

The following document details the Module Interface Specifications for the REVITALIZE app. The REVITALIZE app is an all-in-one health and wellness app, comprised of 1 main section and 3 major subsections. The main section is a calendar which organizes and documents the contents of the 3 subsections. The 3 subsections are the diet section, workout section, and sleep section.

Complementary documents include the System Requirement Specifications and Module Guide. The full documentation and implementation can be found at https://github.com/BillNguyen1999/REVITALIZE/tree/main/docs.

4 Notation

The structure of the MIS for modules comes from Hoffman and Strooper (1995), with the addition that template modules have been adapted from Ghezzi et al. (2003). The mathematical notation comes from Chapter 3 of Hoffman and Strooper (1995). For instance, the symbol := is used for a multiple assignment statement and conditional rules follow the form $(c_1 \Rightarrow r_1 | c_2 \Rightarrow r_2 | ... | c_n \Rightarrow r_n)$.

The following table summarizes the primitive data types used by REVITALIZE.

Data Type	**Notation**	**Description**
character	char	a single symbol or digit
integer	$\mathbb{Z}$	a number without a fractional component in (-∞, ∞)
natural number	$\mathbb{N}$	a number without a fractional component in [1, ∞)
real	$\mathbb{R}$	any number in (-∞, ∞)
boolean	bool	value can be True (1) or False (0)
user	User	represents user object, for users of REVITALIZE
date	Date	represents date object, which is useful to add/set/manipulate dates

The specification of REVITALIZE uses some derived data types: sequences, strings, and tuples. Sequences are lists filled with elements of the same data type. Strings are sequences of characters. Tuples contain a list of values, potentially of different types. In addition, REVITALIZE uses functions, which are defined by the data types of their inputs and outputs. Local functions are described by giving their type signature followed by their specification.

5 Module Decomposition

The following table is taken directly from the Module Guide document for this project.

Level 1	**Level 2**
Hardware-Hiding	
Behaviour-Hiding	Input Parameters Output Format Output Verification Temperature ODEs Energy Equations Control Module Specification Parameters Module
Software Decision	Sequence Data Structure ODE Solver Plotting

Table 1: Module Hierarchy

6 MIS of Main Menu

6.1 Main Menu Module

6.2 Uses

react
react-native
globalStyles: CSS file to change designs of project
Ionicons: Library for icons
Moment Library is used for Dates (ex. setting date formats (YY/MM/DD))
useRoute react file that is used to navigate between screens of project

6.3 Syntax

6.3.1 Exported Constants

6.3.2 Exported Types

MainScreen = this

6.3.3 Exported Access Programs

Name	**In**	**Out**	**Exceptions**
displayDietScreen	User, Date		
displayExerciseScreen	User, Date		
displaySleepScreen	User, Date		
displayCalendarScreen			

6.4 Semantics

6.4.1 State Variables

user: User

date: Date

6.4.2 Environment Variables

dateText: Text object that displays the selected date.

dateButton: Button object that displays Calendar Screen when clicked.

forwardButton: Button object that displays the next day from current Date value in dateText when clicked

backwardButton: Button object that displays the previous day from current Date value in dateText when clicked

dietButton: Button object that displays Diet Screen when clicked

exerciseButton: Button object that displays Exercise Screen when clicked

sleepButton: Button object that displays Sleep Screen when clicked

6.4.3 Assumptions

N/A

6.4.4 Access Routine Semantics

displayDietScreen(user, date):

- transition: Navigates to Diet Screen when dietButton is pressed
- exception: None

displayExerciseScreen(user, date):

- transition: Navigates to Exercise Screen when exerciseButton is pressed
- exception: None

displaySleepScreen(user, date):

- transition: Navigates to Sleep Screen when sleepButton is pressed
- exception: None

displayCalendarScreen():

- transition: Navigates to Calendar Screen when dateButton is pressed
- exception: None

6.4.5 Local Functions

forwardSetDate():

- transition: date.value := date.value + 1. Sets the next day from the current Date value in dateText when clicked.
- exception: None

backwardSetDate():

- transition: date.value := date.value - 1. Sets the previous day from the current Date value in dateText when clicked.
- exception: None

7 MIS of Calendar

7.1 Calendar Module

7.2 Uses

react
react-native
globalStyles: CSS file to change designs of project
react-native-calendars: Library useful for implementing calendars in react-native
useRoute react file that is used to navigate between screens of project

7.3 Syntax

7.3.1 Exported Constants

7.3.2 Exported Types

CalendarScreen = this

7.3.3 Exported Access Programs

Name	**In**	**Out**	**Exceptions**
onDayPress			
onMonthChange			
onPressArrowLeft			
onPressArrowRight			

7.4 Semantics

7.4.1 State Variables

date: Date

7.4.2 Environment Variables

monthText: Text object that displays the selected month.

forwardMonthButton: Button object that displays the next month from current month value in monthText when clicked

backwardMonthButton: Button object that displays the previous month from current month value in monthText when clicked

7.4.3 Assumptions

N/A

7.4.4 Access Routine Semantics

onDayCalendar():

- transition: Changes date value to selected date value in CalendarScreen
- exception: None

onMonthChange():

- transition: Changes date.month.value to new date.month.value and monthText will be changed to string value of new date.month.value
- exception: None

onPressArrowRight():

- transition: date.month.value := date.month.value + 1. Sets the next date.month.value from the current date.month.value in monthText when clicked
- exception: None

onPressArrowLeft():

- transition: date.month.value := date.month.value - 1. Sets the previous date.month.value from the current date.month.value in monthText when clicked
- exception: None

7.4.5 Local Functions

N/A

8 MIS of Sleep

8.1 Container Module

8.2 Uses

react
react-native
react-native-reanimated
react-native-redash
Label: Module

8.3 Syntax

8.3.1 Exported Constants

PI := Math (object that provides mathematics functionality and constants)
TAU := 2 * PI
PADDING := 24

8.3.2 Exported Types

N/A

8.3.3 Exported Access Programs

Name	**In**	**Out**	**Exceptions**
DisplayContainer			

8.4 Semantics

8.4.1 State Variables

date: Date

8.4.2 Environment Variables

BedTimeText: Text object that displays the selected bedtime.

WakeUpTimeText: Text object that displays the selected wake up time.

SleepTimeText: Text object that displays the total sleep time.

ArcStartPos: Polar coordinates object representing starting position of circular slider arc. Modifies bedtime when slid.

ArcEndPos: Polar coordinates object representing ending position of circular slider arc. Modifies wake up time when slid.

CircularSliderArc: string literal object representing an arc.

8.4.3 Assumptions

N/A

8.4.4 Access Routine Semantics

DisplayContainer():

- output: display bedtime, wake up time, sleep time, arc starting and ending positions, and circular slider arc
- exception: None

8.4.5 Local Functions

radToMinutes(rad):

- output: rad * 24 * 60 / TAU
- exception: None

absoluteDuration(start, end):

- output: start > end ? end + (TAU - start) : end - start
- exception: None

formatDuration2(duration):

- output: format duration to hours followed by minutes.
- exception: None

FoodT Module

Module

IndicatorT

Uses

None

Syntax

Exported Constants

None

Exported Types

FoodT = {
calories, *#Calories in meal*
name, *#Name of meal*
carbs, *#Carbohydrates in meal*
protein, *#Protein in meal*
fat *#Fat in meal*
}

Exported Access Programs

None

Semantics

State Variables

None

State Invariant

None

Assumptions

None

Considerations

None

9 Diet Log Module

9.1 Uses

react
react-native
globalStyles: CSS file to change designs of project
Ionicons: Library for icons
useRoute react file that is used to navigate between screens of project

9.2 Syntax

9.2.1 Exported Constants

9.2.2 Exported Types

DietLogScreen = this

9.2.3 Exported Access Programs

Name	**In**	**Out**	**Exceptions**
calculateDailyNutrition	seq of FoodT	seq of <String, $\mathbb{R}$>	
removeFood	food: FoodT		

9.3 Semantics

9.3.1 State Variables

foodList: seq of FoodT
totalNutrition: seq of <String, $\mathbb{R}$>
date: Date

9.3.2 Environment Variables

addFoodButton: Button object that shifts user to Search or Add Food Module.

deleteFoodButton: Button object that deletes a food entry from this module when clicked

editFoodButton: Button object navigates to Log Meal Module

9.3.3 Assumptions

N/A

9.3.4 Access Routine Semantics

calculateDailyNutrition():

- transition: totalNutrition :=$\sum$ foodList
- exception: None

removeFood(food: FoodT):

- transition: foodList := {foodList} \ food. Sets the next date.month.value from the current date.month.value in monthText when clicked
- exception: None

9.3.5 Local Functions

N/A

10 Search or Add Food Module

10.1 Uses

react
react-native
globalStyles: CSS file to change designs of project
useRoute react file that is used to navigate between screens of project

10.2 Syntax

10.2.1 Exported Constants

10.2.2 Exported Types

DecisionScreen = this

10.2.3 Exported Access Programs

None

10.3 Semantics

10.3.1 State Variables

None

10.3.2 Environment Variables

searchRecipeButton: Button object that shifts user to Search Recipe Module.

addCustomMealButtom: Button object that shifts user to Log Meal Module

10.3.3 Assumptions

N/A

10.3.4 Access Routine Semantics

calculateDailyNutrition():

- transition: totalNutrition $:= \sum$ foodList
- exception: None

removeFood(food: FoodT):

- transition: foodList := $\{$foodList$\} \setminus$ food. Sets the next date.month.value from the current date.month.value in monthText when clicked
- exception: None

10.3.5 Local Functions

N/A

11 Custom Meal Module

11.1 Uses

react
react-native
globalStyles: CSS file to change designs of project
react-native-calendars: Library useful for implementing calendars in react-native
useRoute react file that is used to navigate between screens of project

11.2 Syntax

11.2.1 Exported Constants

N/A

11.2.2 Exported Types

CustomMealScreen = this

11.2.3 Exported Access Programs

Name	**In**	**Out**	**Exceptions**
saveCustomMeal	foodInfo: seq of String	FoodT	

11.3 Semantics

11.3.1 State Variables

foodList: seq of String

11.3.2 Environment Variables

addButton: Save food information.

11.3.3 Assumptions

N/A

11.3.4 Access Routine Semantics

saveCustomMeal(foodInfo):

- transition: foodList := foodList ∪ FoodT(foodInfo)
- exception: None

11.3.5 Local Functions

N/A

12 Search Recipe Module

12.1 Uses

react
react-native
globalStyles: CSS file to change designs of project
Ionicons: Library for icons
useRoute react file that is used to navigate between screens of project

12.2 Syntax

12.2.1 Exported Constants

N/ A

12.2.2 Exported Types

SearchRecipeScreen = this

12.2.3 Exported Access Programs

Name	**In**	**Out**	**Exceptions**
searchRecipe	filterList: seq of <String, String>	Seq of <String, String>	
returnRecipeList		Seq of <String, String>	

12.3 Semantics

12.3.1 State Variables

filterList: seq of <String, String>
recipeList: Seq of <String, String>

12.3.2 Environment Variables

searchRecipeButton: Button object that shifts user to Search Results Module, calls searchRecipe().

12.3.3 Assumptions

N/A

12.3.4 Access Routine Semantics

searchRecipe(filterList):

- transition: Populates state variable recipeList using external API call.
- exception: None

12.3.5 Local Functions

N/A

13 Recipe Results Module

13.1 Uses

react
react-native
globalStyles: CSS file to change designs of project
react-native-calendars: Library useful for implementing calendars in react-native
useRoute react file that is used to navigate between screens of project

13.2 Syntax

13.2.1 Exported Constants

N/A

13.2.2 Exported Types

RecipeResultsScreen = this

13.2.3 Exported Access Programs

Name	In	Out	Exceptions
retrieveRecipeInfo	recipeLink: String	<String, FileObject>	

13.3 Semantics

13.3.1 State Variables

recipeList: Seq of <String, String>

13.3.2 Environment Variables

13.3.3 Assumptions

N/A

13.3.4 Access Routine Semantics

retrieveRecipeInfo(recipeLink):

- transition: Retrieve details and picture file of recipe found at recipeLink
- exception: None

13.3.5 Local Functions

N/A

14 Recipe Details Module

14.1 Uses

react
react-native
globalStyles: CSS file to change designs of project
Ionicons: Library for icons
useRoute react file that is used to navigate between screens of project

14.2 Syntax

14.2.1 Exported Constants

14.2.2 Exported Types

RecipeDetailsScreen = this

14.2.3 Exported Access Programs

N/A

14.3 Semantics

14.3.1 State Variables

recipeDetails: seq of <String, FileObject>

14.3.2 Environment Variables

addRecipeButton: Add recipe to Daily Food Log.

14.3.3 Assumptions

N/A

14.3.4 Access Routine Semantics

14.3.5 Local Functions

N/A

References

Carlo Ghezzi, Mehdi Jazayeri, and Dino Mandrioli. *Fundamentals of Software Engineering*. Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition, 2003.

Daniel M. Hoffman and Paul A. Strooper. *Software Design, Automated Testing, and Maintenance: A Practical Approach*. International Thomson Computer Press, New York, NY, USA, 1995. URL http://citeseer.ist.psu.edu/428727.html.

15 Appendix

[Extra information if required —SS]