Use Cases

Use Case 1: Add User

Primary Actor: User

StakeHolder and Interests:

User: - User Wants to be able to add a User in the App.

Precondition: None

Success Guarantee: The user adds another user to the app, using the app.

Main Success Scenario:

1. User opens the app
2. User navigates to the “Profiles” tab,
    a. User selects the button labeled “New Profile”
3. New Page is Open with text boxes labeled
    a. Weight, Height, Name,
    b. Gender CheckBox
4. User the button labeled “Save”
    a. User is added to the list of users.
    b. User is returned to “Profile Info” tab

Extensions:

<All steps> Application crashes.

a1. App crashes

a2. Information is not saved and the user restarts at Step 1.

3.1 User did not fill in the “Name” “Height” or “Weight” textbook

3.2 The application does not allow you to save.

Use Case 2: Change Current User

Primary Actor: User

StakeHolder and Interests:

User: - User changes which user is Active

Precondition: “Add User”

Success Guarantee: The user Changes who is active in the app, using the app.

Main Success Scenario:

1. User opens the app
2. User navigates to the “Profiles” tab,
    a. All Users Profiles are displayed.
    b. User, clicks on a Profile “Name”
3. User selects the Check-Box labeled “Profile for metering”
    a. User clicks “save” or “back.png”
4. User is returned to the “Profiles” tab
    a. The “current.png”, is placed on the Profile that was selected.

Extensions:

<All steps> Application crashes.

a1. App crashes

a2. Information is not saved and the user restarts at Step 1.

Use Case 3: Deleting User

Primary Actor: User

StakeHolder and Interests:

User: - User wants to be able to delete(s) a user's Profile, using the app.

Precondition: None

Success Guarantee: The user deletes another user from the app.

Main Success Scenario:

1. User opens the app
2. User navigates to the “Profiles” tab
    a. All User(s) Profile(s) are displayed.
3. User Clicks on a Profile
4. User Clicks on “Delete Profile” Button
    a. User is returned to the “Profiles” tab, the selected Profile is absent

Extensions:

<All steps> Application crashes.

a1. App crashes

a2. Profile is not deleted and the user restarts at Step 1.

Use Case 4: Updating User Profile

Primary Actor: User

StakeHolder and Interests:

User: - User Wants to be able to update an existing profile.

Precondition: None

Success Guarantee: The user successfully updates a pre-existing profile, using the app.

Main Success Scenario:

1. User opens the app
2. User navigates to the “Profiles” tab
    a. A dropdown with all User(s) Profile(s) are displayed.
3. User clicks on a Profile, in the dropdown tab
    a. The Profile information is displayed.
4. User Updates the Name, Weight or Height textboxes.
5. User select the button labeled “save”
    a. App is updated with the new information

6. User is returned to the “Profile Info” tab

Extensions:

<All steps> Application crashes.

a1. App crashes

a2. Information is not updated and the user restarts at Step 1.

<4a> The updated information is empty

<4a.1> The app does not allow the user the current changes.

Use Case 5: Using Radotech

Primary Actor: User

StakeHolder and Interests:

User: - Use updates the current profile with Radiotech readings.

Precondition: None

Success Guarantee: Users profile is successfully updated with Radiotech readings.

Main Success Scenario:

1. User Opens App
2. User navigates to the “Measure” tab
3. User Clicks “Measure Now”
4. User places sensor in the first 1 of 24 positions
    a. Reading was measured as the Graph is updated.
    b. User Clicks on the button next to “Now Measuring: “
5. Follow 7, until all positions 24 positions are complete
6. App is updated with all the sensor readings to the current profile.

Optional:

1a: User follows, Use Case 2 “Change Current User”

1a.1: User resumes Use Case 5 at Step 2

Extensions:

<All steps> Application crashes.

a1. App crashes

a2. Readings are not saved, User restarts at step 1

2a: Radotech is Off, user is prompted by a Message Box

2a.1: User turns on “Radotech”

2a.2: User continues to step 3

4a. User Turns off Radotech during “Reading”

4a.1 The readings that were displayed, are not saved

4a.2 User Restarts at Step 3.

4b. Radotech battery is at 0%,

4b.1 User is prompted by a message Box

4b.2 User charges batter,
4b.3 Readings that were displayed are not saved
4b.3 User restarts at Step 3.

4c. User simulates all readings at once
4c.1 User selects the simulate readings checkbox
4c.2 App generates readings
4c.3 User is placed in Step 6

Use Case 6: Display Reading
Primary Actor: User
StakeHolder and Interests:
User: - User wishes for to view the readings
Precondition: None
Success Guarantee: Users have viewed the readings.
Main Success Scenario:

1. User Opens App
2. User navigates to the “History” tab
    a. User Select a reading
3. Bar Graph is displayed
    a. User views the readings.
4. User Selects the “Back.png” button

Optional:
3a: User wishes to view a different Graph
3a.1: User Selects the “Circle” or “Body” tab
3a.2: User resumes from Step 4.

Extensions:
<All steps> Application crashes.
a1. App crashes
a2. User restarts at step 1

Use Case 7: Recommendation Tab
Primary Actor: User
StakeHolder and Interests:
User: - User wishes to get recommendation and the Doctor/Specialist recommendation
Precondition: None
Success Guarantee: Users have viewed the Recommendation
Main Success Scenario:

1. User Opens App

2. User navigates to the “Recommendation” tab
3. App, generates recommendation based on the User, poor readings
    a. User inputs the Doctor/Specialist recommendation into the textbox
    b. User Clicks Save
4. User Selects the “Back.png” button

Optional:

3a: User wishes to clear recommendation textbox

3a.1: User Selects “cancel” button

3a.2: User Selects “Save” Button

3a.3 The textbook is updated as blank

Extensions:

<All steps> Application crashes.

a1. App crashes

a2. User restarts at step 1

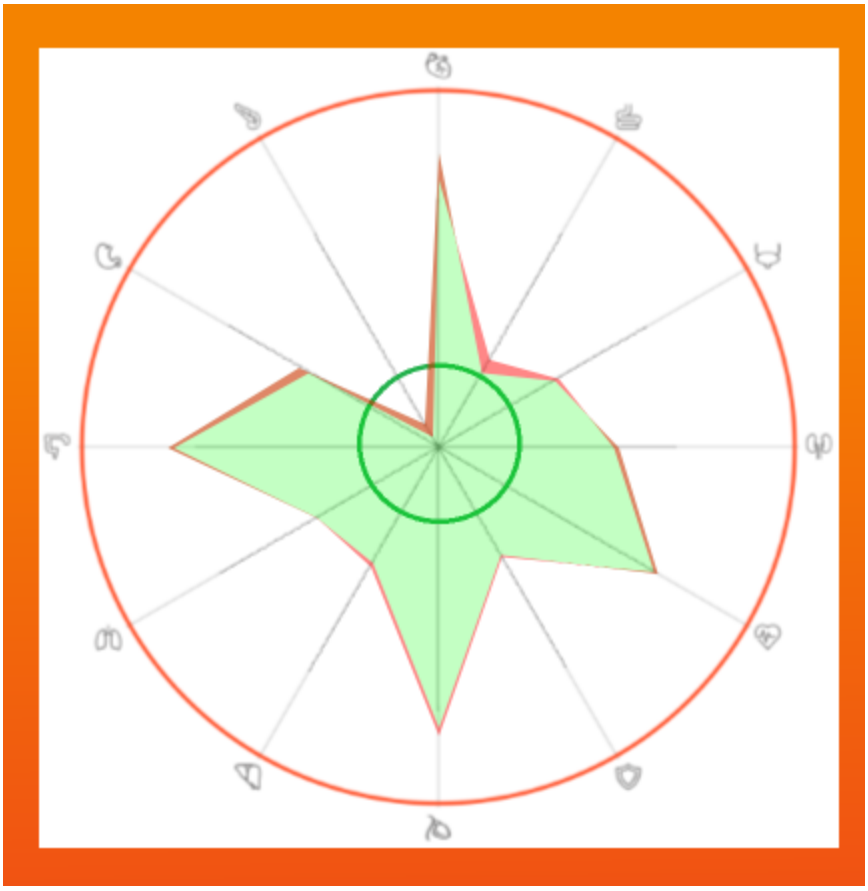
3a. User does not Click “Save” button

3a1. The recommendation text in the textbox is not saved

Textual explanation and Design Patterns

Our Implementation of skin on and off is unique, we assumed the Radotech widget would not record a reading if the device is not on the skin. Our implementation is that you only press the button when the widget is on the skin. There is no chance of getting a reading of zero, as we always assumed the device is on the skin.

Circle Chart has a minimum reading amount, this prevents you from getting a jagged and off looking graph. The radotech app also has this implementation, we believe it is important to reiterate our own understanding of why we manipulate the readings. You could argue it is better to get the raw data then, get a manipulated graph but you can see the raw numbers in the “Chart” tab in the “History” Tab. The point of the Circle tab is to easily compare your left and right side reading. Here is an example to show.



With the pancreas reading being close to 0,0 the user may get a false sense of worry. When the readings for the pancreas simply on average returns lower when compared to the heart or intestines readings.

Design Patterns

We used the Singleton design pattern, as we value not having global variables, we instead shared data between classes, by using friend class attributes along with getters and setters where applicable. This allowed us to easily access data without having the overhead of static global variables.