Darrell Chism

Southern New Hampshire University

CS-499

1-4 ePortfolio Selection and Refinement Plan \*REVISED\*

**Category One: Software Engineering/Design**

* Artifact name, origin, and category

Mobile App from CS-360, Software Engineering/Design.

* Enhancement plan

My enhancement plan is to redesign the mobile app. When I turned in this app as my final project for CS-360, the app worked, but was not visually appealing and did not have everything I had originally planned in the design. I plan on redesigning the app to be more user friendly and more appealing, while also integrating the rest of the functions I had originally designed. Below is my pseudocode:

Function registerUser(username, password):

If username already exists:

Display error message: "Username already taken."

Else:

Create new user account with provided username and password

Display success message: "Account created successfully."

Main:

Prompt user for username and password

Call registerUser(username, password)

Function logWeight(user, weight, date):

Create new weight entry with user, weight, and date

Display success message: "Weight logged successfully."

Main:

Prompt user to log weight for the day

Input weight and date

Call logWeight(user, weight, date)

Function sendPushNotification(user, message):

Initialize push notification system

Send message to user's device

Function scheduleDailyNotification(user):

Set daily reminder to log weight for the user

Call sendPushNotification(user, "Don't forget to log your weight today!")

Main:

For each registered user:

Call scheduleDailyNotification(user)

Function generateWeightChart(user):

Retrieve user's weight entries

Sort weight entries by date

Plot line graph with dates on X-axis and weights on Y-axis

Main:

Prompt user to view weight chart

Call generateWeightChart(user)

* Skills illustrated and course outcome(s) these skills align to.

In the realm of Software Design and Engineering, I'm focusing on giving the app a complete makeover to make it more visually attractive and user-friendly. I'm taking a user-centered approach, carefully considering how people interact with the app to ensure a smoother experience. The revamped app not only looks better but also communicates more effectively. I'm striving to make sure that users of all backgrounds can understand and enjoy it, addressing outcomes related to communication, user-centered solutions, and crafting effective computing solutions.

C**ategory Two: Algorithms and Data Structures**

* Artifact name, origin, and category

Mobile app from CS-360, Algorithms and Data Structures

* In the original design, I wanted to implement a visual chart for users to be able to see how their weight has fluctuated over a period of time. I was not able to achieve this in time, as I was struggling to accurately pull data from my database and input it into chart form. The following is my pseudocode for this:

Function generateWeightChart(userWeightData):

sortedWeightEntries = SortEntriesByDate(userWeightData)

InitializeChartCanvas()

for entry in sortedWeightEntries:

xCoordinate = CalculateXCoordinate(entry.date)

yCoordinate = CalculateYCoordinate(entry.weight)

DrawPoint(xCoordinate, yCoordinate)

for i from 0 to length(sortedWeightEntries) - 2:

DrawLine(sortedWeightEntries[i], sortedWeightEntries[i+1])

AddAxisLabelsAndMarkings()

DisplayChart()

Function SortEntriesByDate(entries):

// Use a sorting algorithm (e.g., merge sort) to sort entries by date

// Return the sorted entries

Function CalculateXCoordinate(date):

// Calculate the x-coordinate based on the date and chart dimensions

// Return the calculated x-coordinate

Function CalculateYCoordinate(weight):

// Calculate the y-coordinate based on the weight and chart dimensions

// Return the calculated y-coordinate

Function DrawPoint(x, y):

// Draw a point or mark at the specified (x, y) coordinates on the chart

Function DrawLine(entry1, entry2):

// Draw a line segment connecting two entries on the chart

Function InitializeChartCanvas():

// Initialize the chart canvas or graphical component

Function AddAxisLabelsAndMarkings():

// Add labels, titles, grid lines, and other annotations to the chart

Function DisplayChart():

// Display the generated chart within the app's user interface

Main:

userWeightData = RetrieveWeightDataFromDatabase()

generateWeightChart(userWeightData)

* Skills illustrated and course outcome(s) these skills align to.

Within the Algorithm and Data Structures domain, my focus is on optimizing the app's performance and functionality. Through algorithmic enhancements and efficient data structure utilization, I manage trade-offs in design choices to ensure optimal problem-solving, directly addressing the course outcome related to designing and evaluating computing solutions. By rigorously evaluating computing solutions and incorporating industry-standard practices, I meet course outcomes tied to algorithmic principles and implementing computer solutions. Furthermore, by considering security implications and anticipating vulnerabilities in design, I cultivate a security mindset that safeguards the privacy and integrity of data, aligning with a holistic approach to software architecture and addressing the security mindset outcome.

**Category Three: Databases**

* Artifact name, origin, and category

Mobile app from CS-360, Databases

* Enhancement plan, including pseudocode or flowchart of the design

Building upon my initial design, my goal is to create a seamless experience for users when recording and visualizing their weight entries. This enhancement plan includes refining the database schema to ensure distinct and accurate weight entries. I'll implement robust data validation checks to prevent erroneous inputs and enable user-friendly feedback. Moreover, the pseudocode outlines mechanisms for handling different scenarios, such as updating existing entries or recording new ones, based on date comparisons. I'll also optimize data retrieval for generating user-centric weight charts and ensure secure user authentication and access controls. The following is my pseudocode:

Function authenticateUser(username, password):

hashedPassword = FetchHashedPasswordFromDatabase(username)

if hashedPassword is not null:

if VerifyPassword(password, hashedPassword):

return true

return false

Function logWeight(userID, weight, date):

if ValidateWeight(weight):

existingEntries = QueryWeightEntriesFromDatabase(userID)

if existingEntries is empty:

InsertWeightEntryIntoDatabase(userID, weight, date)

DisplaySuccessMessage("Initial weight entry recorded successfully.")

else:

latestEntry = GetLatestWeightEntry(existingEntries)

if latestEntry.date == date:

UpdateLatestWeightEntry(existingEntries, weight)

DisplaySuccessMessage("Weight entry for " + date + " updated successfully.")

else:

InsertWeightEntryIntoDatabase(userID, weight, date)

DisplaySuccessMessage("Weight entry for " + date + " recorded successfully.")

else:

DisplayErrorMessage("Invalid weight input. Please enter a valid weight.")

Function retrieveUserWeightData(userID):

weightData = QueryWeightEntriesFromDatabase(userID)

return weightData

Function generateWeightChart(userID):

userWeightData = retrieveUserWeightData(userID)

sortedWeightEntries = SortEntriesByDate(userWeightData)

InitializeChartCanvas()

for entry in sortedWeightEntries:

xCoordinate = CalculateXCoordinate(entry.date)

yCoordinate = CalculateYCoordinate(entry.weight)

DrawPoint(xCoordinate, yCoordinate)

for i from 0 to length(sortedWeightEntries) - 2:

DrawLine(sortedWeightEntries[i], sortedWeightEntries[i+1])

AddAxisLabelsAndMarkings()

DisplayChart()

Main:

InitializeDatabaseConnection()

username = GetUserInput("Enter username: ")

password = GetUserInput("Enter password: ")

if authenticateUser(username, password):

userID = FetchUserIDFromDatabase(username)

loggedIn = true

else:

DisplayMessage("Authentication failed.")

loggedIn = false

if loggedIn:

DisplayMessage("Welcome, " + username + "!")

weight = GetUserInput("Enter your weight: ")

date = GetCurrentDate()

logWeight(userID, weight, date)

generateWeightChart(userID)

else:

DisplayMessage("Access denied.")

CloseDatabaseConnection()

* Skills illustrated and course outcome(s) these skills align to.

In the realm of Database management, I will be enhancing how the app handles data. To showcase data validation and integrity, I will be implementing error handling mechanisms and refining the database schema. Also, to showcase a security mindset and to demonstrate addressing potential vulnerabilities, I will be implementing privacy safeguards and secure password hashing. This will also showcase my code fulfilling industry-specific standards and goals.

**ePortfolio Overall**

* Skills/outcomes planned to be illustrated in the code review

Algorithmic thinking

Data structures

SQL proficiency

Integration of user authentication

* Skills/outcomes planned to be illustrated in the narratives

Effective communication of technical concepts

Understanding and addressing user needs

Iterative development and feedback incorporation

Reflecting on challenges and learning experiences

* Skills/outcomes planned to be illustrated in the professional self-assessment

Self-awareness of technical strengths and weaknesses

Continuous learning

Skill enhancement

Goal setting

Planning and time management skills

Capacity for collaborative work and team interaction

Communication and documentation skills

Reflection on personal and professional growth

**Questions about ePortfolio**

* After reviewing my enhancement plans, are you okay with me using the mobile app for all three categories? If not, which categories are you okay with me using the mobile app to showcase enhancements?