Robert Petrelli **12/7/18**
CMPT 220L **Prof. Juan Arias**

Abstract:

The Alcohol Consumption Simulator (ACS) is a program that operates visually within an external user interface and utilizes several user inputted values and variables to determine the simulated subjects BAC levels and symptoms as well as any warnings that are advised based on their condition.

Introduction:

The ACS originated from my desire to code something UI based from my knowledge of Java from high school. Basing it on alcohol and its different symptoms was something that I selected based on its quantitative founding and educational use.

The following sections of this paper will detail:

- How the ACS functions code wise
- How users are intended to interact with the ACS
- How the classes interact in UML
- The specific details of the problem the ACS is addressing
- Other alternatives to the ACS
- The user manual for the ACS
- The goals accomplished by the ACS
- References used to create the ACS

Detailed System Description:

The ACS takes the users inputted information to generate a simplistic human stand-in and applies the information given to identify how drunk the simulated human is. The user is prompted to input the simulated human's gender, # of drinks consumed of beer, # of drinks consumed of wine, # of drinks consumed of wine, hours in which the above drinks were consumed, whether the simulation had eaten, whether the simulation had drank water, whether the simulation had slept, and whether the simulation was on medication. The ACS then compiles this information against a series of if statements and switch statements to output the appropriate message and warnings based on the answers entered.

Users are intended to begin the program with little to no information already known. All of the input fields and buttons are either programmed to certain values or only return a single pop-up. Beginning on the start menu, users are faced with buttons labeled Start, Help, and Learn More. Pressing stop opens up a new content pane with the true ACS, pressing Help will engage a pop-up with more information and instructions, and pressing Learn more will produce a pop-up with more information and sources.

After pressing start and opening the true ACS window, users will have the ability to enter the information for the simulation, or press buttons labeled Back, Details, What is a 'Drink' , and

Submit. Pressing back will return the user to the start menu, pressing Details will produce a pop-up with details about the ACS and its information, pressing What is a ‘Drink’ will create a pop-up informing the user about the common unit of a drink which is the same unit of alcohol across separate beverages. Pressing Submit will begin the logic processing of the ACS which will take the users values from the buttons and text fields in the window and produce a pop-up corresponding to those values.

Main
+main(): String[]

Start
- startJButton: JButton - helpJButton: JButton - sourceJButton: JButton - TitleJLabel: JLabel - Base: base

+getBase()
+Start()
-createUserInterface()

Base
-drinkJButton: JButton -submitJButton: Button - detailJButton: Button - backJButton: Button +beerNum: int +liquorNum: int +wineNum: int +drink: int +timeNum: int -maleString: String -femaleString: String -noFood:String -noWater:String -noSleep: String -drugString:String +food: Boolean +water: Boolean +sleep: Boolean +Drug: Boolean +gender: Boolean

+Base()
-createUserInterface
+setDefaultCloseOperation(exitOnClose: int)

Requirements:
The specific problem that the ACS solves, determining BAC from several influencing factors, requires certain specific details in order to function. The first detail or assumption that the ACS functions on is that besides the variable available to the user, everything else is assumed constant. The second detail is that the simulated human is assumed to perfectly average. Males are assumed to be 21 years old and 200 lbs., while females are assumed to be 21 years old and 170 lbs. The third specific detail is that every drink consumed, and input conforms to the US standard unit of a ‘Drink’

Literature Survey:
A similar program that was developed is marketed as AlcoholEdu, and is sold as a product by Everfi, as “Alcohol Awareness, Prevention and Training for College Students.” This program operates within a browser and functions as an educational tool to inform students about the dangers and effects of alcohol.

Another work that attempted to solve this problem was the Alcohol Help Center’s Blood Alcohol Calculator. This calculator takes the input of gender, weight, drinks consumed, and time to produce a BAC result. This program also acted as a source for this project but neglected to introduce the effects of hunger, dehydration, tiredness, and use of medication, as the ACS does.

User Manual:
The program, once launched, allows the user to select any of the three buttons labeled Learn More, Help, and Start. Once Start is pressed the ACS will be on screen. The user at this point should input the values they want simulated on a human body.

For Example:
Gender: Male
# of ‘Drinks’ of Beer: 2
# of ‘Drinks’ of Wine: 1
# of ‘Drinks’ of liquor: 1
# of hours drinks were consumed: 1
Ate Recently: Check
Drank Water Recently: Check
Slept well: Check
On Medication: Unchecked

Output:

“After consuming 4 drinks over the period of 1 hours, your BAC as an average male would be around 0.069 grams per milliliter of blood. You are feeling a bit more relaxed and might have a feeling of well being or euphoria. It is currently a poor idea to drive. You will return to sobriety in about 5 hours.”

After receiving the output, the user has the ability to accept the pop-up and re-do the simulation under different simulated conditions.

Conclusion:

The goals of this project in the proposal were to
“create a program that will allow the user to simulate the effects of different types and amount on the human body. The user will be able to select the type of alcohol, how much is ingested, over what period the ingestion took place, the simulated body’s status before drinking (recent food intake, hydration, sleep, etc.), and whether the simulated human body is an average college age male or average college age female.”

In this regard I believe the project accomplished all of its stated goals. However, in my proposal I did also list stretch goals that would be add depending on time and prior feature success. “allowing the user to select from multiple body weights and heights as well as age for both males and females.” This feature did not make into the finished build as it would have taken too much time and effort and may not have been function by the due date.

References/Bibliography:

“Blood Alcohol Calculator” Accessed November 24th, 2018.
http://www.alcoholhelpcenter.net/Program/BAC_Standalone.aspx

“How to Use Buttons, Check Boxes, and Radio Buttons” Accessed November 5th, 2018
https://docs.oracle.com/javase/tutorial/uiswing/components/button.html