COP3503 Final Project: Gainezville
Project_Team 38

Original Files List:

Header Files:	Line count:	Size:
BackupOutput.h	46	2KB
Exercise.h	41	1KB
ExerciseAction.h	40	1KB
healthinfo.h	90	2KB
User.h	89	3KB
Workout.h	36	1KB
Info.h	45	3KB
WorkoutHistory.h	32	679 Bytes

Source Files (.cpp):	Line count:	Size:
BackupOutput.cpp	2122	63KB
Exercise.cpp	24	877 Bytes
ExerciseAction.cpp	69	1KB
healthinfo.cpp	384	10KB
User.cpp	176	3KB
Workout.cpp	161	4KB
Info.cpp	912	34KB
WorkoutHistory.cpp	39	689 Bytes

Total line count *excluding* external libraries: 4306
Total line count *including* external libraries: 7811

Lesson Learned:

- **User Interface/External Libraries:** After doing a lot of research, our team found that SFML would be the most ideal graphics library to use. This component has been, by far, the one with the biggest learning curve for our assignment. SFML's library is not the friendliest and has a lot of compatibility issues with IDE's, C++ versions, and compilers. Once we got it to work, we've been messing with functions within the library to create a pleasant and aesthetic experience for the users.
- **Version control:** Because this project was so extensive and required seven people to constantly update information within the code, version control was an imperative tool for the design of our Gainezville application. Throughout the project, we used GitHub and it was useful as we constantly committed changes and performed rollbacks to previous versions whenever we lost any information throughout the design process.
- **Extensive code base:** For most of our group members, this was the first time we've been tasked to write an entire, extensive code base. Therefore, we learned how to delegate realistic tasks for everyone and create realistic deadlines to produce the best application possible.
- **Collaboration:** Being in a big group of developers is difficult as everyone may have a different way of approaching a solution. Therefore, this project was a great way of learning soft skills such as communication, listening, and considering other people's ideas, thoughts, and solutions to each problem we encountered. Therefore, we created a group chat workspace where we voiced ideas and concerns before committing to solving a problem in a specific way.

Future Work:

- **Use better graphics library:** Although SFML provided us with a reliable solution for creating a user interface, it was not easy for the external library to work on every group member's machine. There are more complex libraries for GUI's in C++, but in the future considering switching over to C# to create a more aesthetic application would not be a bad idea. We also would like to create a compatible application that can be extracted onto iOS or Android, and C#/Swift may be a better solution in that regard.
- **Adding networking/interaction features:** To further upscale our fitness solution and attract more members, we may want to allow members to interact with each other through the means of a blog/comment section in the exercises/nutrition sections. This would incentivize members to post better workout and dieting solutions between one another. This can also be a good opportunity to apply more complex computer science concepts such as machine learning to tailor results to the user's goals.
- **User experience: making features more accessible/pleasing for user:** Like previously mentioned, transitioning the idea onto a language that prioritizes user experience would be ideal for upscaling this project into a meaningful mobile application. To do so, we would consider C# or Swift to develop application for the two most popular operating systems.