**Weightlifting Competition Project Requirements**

**User interface**

Excel was used as the framework for this project. The original work was done on Excel to sort the lift order list as a simple tool to keep track of Olympic weightlifting competitions. The rules are easy to talk about but hard to remember to implement when you must apply the rules during the competition and repeat a mental and physical sort every 20-30 seconds. This sort is made complex by the addition of the “phase” of the competition along with the concept of “who has waited the longest”. The sort rule is”: “snatch before C&J”, “lowest weight first”, “1st attempt before 2nd ,” “2nd before 3rd,” “who is waiting longest”, and “lowest Start number”. As more competitions were held, the spreadsheet program became more and more complex with added macros and improved user interface leading to a lot of spaghetti code and inconsistent style.

Bugs and spaghetti were unraveled over the years as experience demonstrated the challenges and fears of various types of users. The software phobic users were relatively easy to train but users familiar with Excel shortcuts and commands very often did something that confused the macro coding. The ultimate solution was to lock out the mouse or key combination shortcuts that people had learned as far back as Excel 2003. Sure, it is Excel but try at least to make it not act like Excel. Then somehow make people think it wasn’t Excel at all.

Eventually, the Ribbon was customized to eliminate all but the normal file tab. This user Interface work was generally done ad hoc and thus is inconsistent in style and implementation. It became more consistent but there is a lot of Crufty code around that is not used.

Built in documentation was added at the request of Paul Marini. He designed a lot of improvements in the UI and taught me a lot about what competition managers would actually need to do.

**Custom Ribbon**

To have a custom ribbon and to enable developer mode, the XML formatted Excel file package has to be edited. I used “Custom UI Editor For Microsoft Office” to edit the XML file. There may be safer and more modern ways to enable and disable the ribbon available since 2007. A link to the tool I used is: <http://openxmldeveloper.org/blog/b/openxmldeveloper/archive/2006/05/25/customuieditor.aspx> You can really mess this thing up with these tools but you won’t lose too much work if you are systematic (don’t ask me how I learned that.)

The “startFromScratch” true/false variable at the top of each Excel XML file will re-enable the standard ribbon to expose the “Developer” tab, which is needed for opening the VBA editor. In that case, you see both the standard ribbon and the custom ribbon. Note that the key capture in the paragraph above eliminates the traditional F11 shortcut. Make sure you change any ribbon parameters while excel is closed and restart Excel. Various Excel versions ask the enable Macros on load question. These must be enabled for the show to start.

**Simultaneous Execution on a single machine**

Generally, you should not run these programs (Lift Manager, Lift Weigh In and Lift Loader) on the same machine at the same time. Excel naturally wants to run in one instance with different workbooks. The Ribbon changes for each cause some interesting faults when used simultaneously. I did not attempt to programmatically make them compatible. In the old days you could have separate instances of Excel run on the same machine if you held certain keys while launching the second copy. Maybe that still exists but I just moved on with discipline to not run two at the same time on the same machine.

**Extra Code/Features**

One part of the code enables an external website to display a live scoresheet. A XAMPP server was installed on a USB Stick and a version is in this repository. On startup, when Lift Manager detects the USB stick with XAMPP installed, it starts the apache server as a local website. The USB stick volume name should be LIFTDATA and a root directory is XAMPP with everything needed inside. The Web pages and support files are in the htdocs folder in this repository. Lift Manager outputs the “Leaderboard” and “LiftOrder” data in a CSV style file after every lift. The leaderboard.php or liftorder.php pages are served up to any browser that points to the Lift Manager system on the LAN. This LiftData has not been used on any OS newer than Win7… if the process checking changed on later OS releases, this will need some modification.

This was also used to successfully export “live” data to a publicly available website external to the system alongside of a live video stream. Generally, the Liftorder.php and Leaderboard.php files are copied to an external site with the liftorder data delivered to the site via Putty SCP. Lift Manager kicked off the Putty SCP command. To use the external server, you must add your own external software such as putty or equivalent and work out the permissions. These PHP files are nothing special… no CSS or complex coding. They should be redone in a much prettier website but they served the purpose of presenting a readable scoresheet projected on a big screen behind the platform in common venues with cheap projectors.

Lift Manager creates reports after the competition is complete. The main report is in the USAW format (which is emailed to USAW for standard National Reporting requirements) though that sheet could be modified for other international federation reporting requirements. These results are now also modified to include the 2017 tie break rules.

Lift Manager and Lift Weigh In use the IWF calculation and coefficients for the Sinclair formula. They also use the Meltzer-Faber factors for adjusting age in lifters over 30. The Sinclair values are implemented via an equation calculated in each program. The Meltzer-Faber number are a look up table in a hidden sheet. These could be reduced to a formula but I never got around to it.

The IWF almost always has single gender, single age group, and single weight class sessions. The famous Eleiko system maxes out at 18 lifers per session. Lift Complete accommodates mixed everything –gender, ages, and weight classes with an unlimited number of lifters per session. Practical sessions do work up to about 25 lifters but generally only youth sessions where lifters seem to move faster. The most ever tested in competition was 32. This creates a small problem in setting up start numbers according to the IWF regulations. Since Men and Women have 69kg weight classes, with the help of some USAW national referee trainers, I chose a method to make this work (arbitrarily Women before Men) but it is unspecified in the IWF rules. However, we think this has little impact on the majority of competitions.

IWF competitions also commonly have A, B and C groups of lifters in a given class. Lift Complete does not internally keep track of ranking of a B or C group lifter that manages to lift to a level that ranks with A group lifters in a particular competition. In the past, we have used external spreadsheets after the fact to reconcile the final ranking.

Parts of the code are now unused. Inspection will find considerable code related to records checking. At the early phase of the project, the Pacific Weightlifting Association records for each class were maintained on the PWA website. With some screen scraping, one could input the “current” records for each class into Manager (on a hidden sheet) and then Manager would flag a record attempt for the announcer via a modeless popup message. Unfortunately, this only worked in the Northern California area, and eventually, those records were not maintained. Thus, while the controls are now hidden, the unused code remains.

Another part of the code may or may not be used. USAW at one point added youth weight classes that are outside the range of the IWF classes. Those are included in Lift Weigh In but may not be valid in the 2017 quad. However, the new IWF Women’s and Junior’s classes have been added for the 2017 quad.

**Note on OS Versions**

This project originally started on Windows XP and Excel 2007. Big changes happened at Microsoft between Excel 2003 and Excel 2007. And even bigger changes with Win Vista and Win 7 introduction. And then more changes again with 32bit versions and 64bit versions of the OS and Excel. In theory, Lift Complete does detect Windows XP and runs well but it hasn’t been tested since Microsoft deprecated XP. File structure changes did happen between Windows XP and Windows 7. (I skipped Vista). Generally, these have been extensively used on Win 7 through Win 10 and Excel 2007 and 2010. It has been reported that these continue to work all the way through Excel 2016 but my own experience with web based Office 365 has been poor.

Since Lift Manager, Lift Weigh In and Lift Loader are not hard linked, one can use each one separately in a modular fashion. These are linked using a text file transfer mechanism.

The communication is a file written by one program on one computer and a read by another over the network. The file structure/location is then specific to the OS. These files are simple and tiny CSV files.

In about 2013, Microsoft started supporting VBA in Excel on a Mac. The version of Lift Complete in this repository fails on a Mac because the OS detection mechanism and underlying file system does not know MacOS. It may be possible to make small changes to work on a Mac on later Excel versions. Ron de Bruin has many tips on how the Mac version is different. One thing for sure, the Password and locking mechanism is different. A VBA project locked on the PC version cannot be unlocked on a Mac. Even if the file system differences are accommodated, there is a high likelihood of complex OS calls that will fail. No effort has been expended on the mac version. The versions in this repository have had their VBA project passwords removed.

L**ift Complete Professional and Lift Complete Premium**

These two versions of Lift Complete are no longer available. The additions were a separate embedded hardware system with software that managed referee votes for the good lift and bad lift calls. In addition, there was a Windows network app that allowed this data to be displayed on a leader board and in the warm up room. This hardware has had little demand and even less support. The rest of the app for managing the meet is the same as Lift Complete Lite.

**Forks**

There are no known forks so far. If you decide to fork and rework part or all of the system, please let me know for my edification. I don’t care if you do change it but it would be fun to know what you did and why.

**Known Issues**

Lift Manager has one known issue when a set of conditions align and the user has made some errors. If a user presses the wrong button “Good Lift” two times or more, the “Undo” button can only undo the very last press. This causes erroneous data to be recorded in the scoresheet. If the user then clicks on the “Field” where the bad data is recorded, there are certain conditions where the error cannot be fixed and the program crashes. The “Field” is the denoted by the purple box in the user poster document in this repository.

Work Around: record the error on a piece of paper and then correct the data in the field after Snatch session is complete or the C&J session is complete.

**Acknowledgments and Code Quality**

There are large sections of code of high quality originally written by Ron de Bruin and his pals. This is clear from the quality of the Ribbon modifications and the management of the user key combination lockouts. Dana Le provided the complex sort algorithm for the lift order and the original sort triggering mechanism.

All other code is hacked by me. The code is of varying quality and generally follows poor, or at least not good programming practice standards. I apologize for the quality with my only excuse: I learned VBA after not programming anything for 30 years. In that time, whole object oriented programming languages were invented. In addition, the optional apache website and related Php code is the very first web project I ever attempted. I am sure it could be more easily and better done with modern tools and it would look so much better.

Extensive user testing was completed with the tolerance of the Pacific Weightlifting Association over the space of several years. Excellent individual testers with great feedback were Eleanor, Paul, Chioma, Mia, Deana, Seth and a whole host of other high school volunteers. Those pioneers took the arrows in the back while we learned what works, what is important and how to react under the time pressure of real competitions. Through those folks, I learned what UI design really entails. I now have a deep appreciation for the folks that make user interfaces that are intuitive and hide the complexity of what seems to a human as very straightforward logic.