Factom

Making the Factory of Tomorrow

# Project Post-Mortem

# ETC Pittsburgh – Spring 2012

Last Revised: April 30, 2012

## Personnel

### Advisers:

Ruth Comley <rcomley@andrew.cmu.edu>

John Dessler <jdessler@andrew.cmu.edu>

### Team Members:

Zero Liu <xiyuanl@andrew.cmu.edu>
Producer

JJ Hou <jingyinh@andrew.cmu.edu>
Game Programmer

Sherry Ma <qianrum@andrew.cmu.edu>
Database Programmer

Nicholas Sciannameo <nscianna@andrew.cmu.edu>
Sound, Web Content

Albert Tian <weichuat@andrew.cmu.edu>
Lead Programmer

Meng Xie <mxie@andrew.cmu.edu>
3D Modeler and Artist

## Introduction and Overview of the Project

The Factomo project is sponsored by General Motors. Our client representatives, Bob Tilove and Ron Lesperance, are from GM Research and Development.

The original goal, as stated by the clients, of this project, was to create a full-scale simulation experience of an automobile assembly line. This project would range from allowing the player to manipulate part installation order at individual assembly stations, the positioning of tools (while keeping track of ergonomics and worker position), worker comfort criteria (such as music playing and the temperature of the facility), as well as the interactions between workers on the line. Finally, the game would be centered on crowd-sourced cooperation – having many people, possibly in a competition setting, attempt to work off each other's design ideas and layouts; the idea being that this could lead to the creation of a new layout or series of trends that could be used by GM in the real world.

Within this large-scale goal, we chose a small component. Working off what we were able to learn about Fall 2011 semester's project, Sigma, we continued down the track of creating a simplified plant layout experience. This time, however, since the goal was to really focus on the layout itself, rather than worker-manager communication, as was the case with Sigma, we took a different approach.

The core of our product is the plant layout experience. To do this, the player is presented with a simplified series of steps to build GM's En-V concept vehicle. These 11 stations must be placed in a prescribed order, as dictated by the client. Certain parts can be installed in any order, but the main choice the player has is what machine speed to select. To do this, one must consider the production budget as well as production goals to ensure that the decision will in fact increase efficiency. Upon completing a layout, the player enters simulation mode and watches as cars are made. The game then measures the amount of time it takes to create 10 vehicles, extrapolating this information out to a 30-day time frame, which provides a more useful metric. The faster one is able to produce vehicles, for the lowest cost, and with the least floor space consumed, the better the layout is.

As for collaboration, originally we had sought to implement a system to track references between designs. This would allow us to track the usefulness of designs and allow an additional metric for measuring success, but we had to remove it due to a lack of client interest and time limitations. As a result, the ranking system displays the most successful layout currently available in the system for players to continue working on. While we lose the ability to have players work on and attempt to improve an "underdog" layout, we do gain data integrity, because without the ability to track who is referencing what design, it would be very easy for a player to simply make a very minor change and have his name appear as a more successful layout.

## Deliverables

For this project, we have created a Unity WebPlayer based game (a standalone executable will also be compiled), which stores and accesses content in a central MySQL database. There is also a registration website that interfaces with this same database to handle user authentication and to help players learn more about the game as well as manage their account information.

The following is a breakdown of how the system works as currently implemented during development.

Registration website on ETC web server – accesses DB information stored in MySQL on Ubuntu Virtual Machine – sends email to players regarding registration & account management

Unity WebPlayer game stored on Ubuntu VM, calls PHP scripts held on same VM to connect to, read, and modify content from MySQL DB on the VM. (In the case of the executable, it accesses these PHP scripts on the VM and makes the DB changes.)

All of this content will remain at the ETC, managed by IT, but will also be packaged for future implementation elsewhere.

Additionally, we will be supplying a Game Design Document and software documentation, that chronicle how we got to where we are, what features / client requests we addressed, what we didn't address and why, and what feedback we have received, from both playtesters and our clients. They will also outline deploying the products.

## What went well?

This project was a challenge right from the start and the number one issue we had was the large list of requirements or requests provided by the client. That said, we were, in time, able to pretty successfully weed through and prioritize what they wanted and what we felt we were able to do. Overall, the team worked reasonably well together, given the level of frustration provided by a massive scope. By the time we got to half presentations the feature set and idea of the game was mostly set – we knew it was going to be almost exclusively about plant layout and that it was going to be a stripped down version of that. There was no way we would be able to simulate the experience at the level our clients really wanted. In the end, coming up with a final product that is playable and reasonably useful has proven to be an impressive feat, considering the disorder the massive scope initially placed the team in.

Other things that went well include working with our clients and playtesting our product. All things considered, our client was quite easy to work with, excluding the ambiguous project itself, Ron, especially, was very willing and eager to share with us whatever information we requested. He responded quickly and seemed to be genuinely interested in what we were doing. On our trip to Detroit, both Ron and Bob took significant time out of their schedules to meet with us and give us a tour of an active assembly facility. While this was slightly embittered by the general lack of direction of the project, it is safe to say we are all pleased with what we have been able to accomplish and the relationship we have all been able to cultivate.

Our playtesting sessions also went remarkably well. We gained highly valuable feedback from all of the tests arranged and many of our player's suggestions were incorporated directly into the product, especially when it comes to UI and tutorial elements. Finally, we were also pleased with being able to have some engineering industry workers examine the product and deem that they do see potential in attempting to bridge the gap between simulation game and simulation software.

## What went wrong?

Many things. Our top issue this semester was communication. Whether it involved the client, scheduling, or keeping our advisers in the loop, there was a thorough breakdown throughout the semester. Most likely, this issue started when the problem of scope really became a main issue. It is unlikely that any of us anticipated the indecisiveness on the client's behalf or the number of decisions that would be left completely up to us. Instead, we anticipated a more complete idea from the client and unfortunately we were faced with an abstract concept from which to try and create something. Discussions with the client turned into something closer to negotiations early on and while that sentiment eventually disappeared as production began, it tarnished what could have been a better and more collegial relationship throughout the project. It is likely that, to some extent, this was also a carryover of last semester's project as well, so we kind of had to clean up whatever impressions they had of the last team's abilities before we could introduce our own. This is something that we should have tried to forecast and deal with in advance.

Overall, an abstract idea would have been fine, but what we received was more like an abstract idea with an exceptional list of requirements. This was not particularly helpful and while it may have looked like they presented us with a viable idea, it instead became so large that it took us many weeks just to get our heads around. Adding this to our tour of a GM assembly facility, we realized all that they envisioned and were collectively blown away by the proposition.

When it came to scheduling, a lot of the issues revolved around not having a firm grasp on what the end product was going to be, because we were unable to really figure out what the client was looking for or would be satisfied by until well into the semester. Ultimately, it is unlikely that this really changed the outcome of the project, but it did certainly lead to a bit of floundering when it came to answering what we were going to do next or how we were going to approach something that was not started yet. We were probably saved by our various skillsets and perseverance as far as being able to address problems as they came up, but it was less than ideal.

Keeping our advisors in the loop proved to be a challenge in part because of a lack of a defined schedule. As a result, they were unable to efficiently head off issues in advance. There was also the situation of a language barrier. This made it difficult to quickly and accurately ascertain the status of any component of the project. While not necessarily a huge problem, it did result in some collective hesitancy to either ask about the project in depth or unilaterally reveal project material on the part of team members.

Quarters presentations were farce-esque, in part due to a lack of any real tangible progress on our parts, but also due to a fair bit of negativity from faculty/staff when it came to this client and project overall. We did not anticipate this, and expected a clean slate from them, as all but one of the team members was new to the project, but this was not the case. As a result, while we did get some good advice, such as Jesse's suggestion to immediately start prototyping (which we did), there was also great concern about the client's intentions and expectations for the project. We were already frustrated by the project at this point and this added tentativeness from the ETC-side was not particularly helpful or insightful, when we could only address what we had done with the project and client up to that point in the semester. While it is important to leave as much open to the team members as possible, as far as dealing with the client, dictating what we would be able to accomplish, etc., some sort of baseline should have been established at quarters or previous to quarters by faculty/staff with the client if there was a genuine concern, which there seemed to be.

## Lessons Learned

The top thing we learned was to define as soon as possible, for the client, what is and is not possible, closing doors and limiting the project as much as possible well in advance of beginning production. It would have been a better case scenario if we had been able to do this, then adding more features if we had time. Unfortunately, we were so collectively daunted that attempts to prioritize features, combined with pushback from the client, resulted in wheel-spin until we had enough of a time crunch that we had no choice but to lock-in and go, addressing what we could, when we got to it. This was obviously not a good situation, and contributed to a long list of issues, most of which were outlined above.

Had this process occurred more properly, it would have likely allowed for more conventional ways of scheduling and task management to be used as well as for more information to be exchanged between the clients, advisers, and team members.

Having a remote client with tech barriers is an issue that should be addressed early on and will consume an inordinate amount of time. In the end, we were able to have the game work over their network, but because of a lack of IT support on their end, we were prevented from having them go hands-on with a product until way at the end of the semester. Even with technical issues, had they been closer we would not have had to worry about this, as much, and it would not have had a negative impact on the client's impression of the team.

## Conclusion

We do not know what the client's final plans for this project will be, but we do plan on having it be accessible, with servers running, at the ETC, until it is determined that it is no longer of use. The server setup is fairly straightforward with no really special configuration required so there is limited concern about that. Finally, the product runs well in Unity WebPlayer – a plugin that is fully cross-platform, so there is limited concern about compatibility. Our clients have also expressed an interest in having the deployment information sent their way, for possible use internally, which it will be.

Overall, this project, while frustrating, also had many rewarding moments. The final product attempts to hit as many requests as possible, while still being playable and stable. In the end, given the relative disorganization this semester started with the end result is impressive and a very good platform to attempt to build off of in the future.