Hans Brown

Paul Dreger

Cara Fisher

Deema Al-Shoaibi : Canadiengineers

12-15-16

“Wizard Needs Matches” Maintenance Plan

After getting the green light from Valve, discussing the pricing and Valve’s revenue cut, and the deployment of Wizard Needs Matches on the Steam store, the game will need continual maintenance in order for it to become successful.

Software maintenance can be divided into four types: corrective maintenance, adaptive maintenance, perfective maintenance, and preventative maintenance. Corrective maintenance involves correcting faults in the software and/or hardware. In terms of the project, this would mean going through the list of bugs we ourselves prepared and other bugs that may be brought to our attention by players of the game. Adaptive maintenance will probably be needed the least for the first year, because it is unlikely the environment will change so drastically that our product will be rendered inoperative. Perfective maintenance is for polishing or improving the product in some way. For example, carrying out every idea that is on the product backlog for the project like enabling multiple players. Preventative maintenance is, just like it sounds, maintenance performed to prevent foreseeable problems in the future. Improving the modularity, or even the documentation, of our project might be considered preventative maintenance since both will make the code much easier to read, understand, and provide a quick fix for if a major issue arises or edit to incorporate new content. In that case, using an object-oriented language like C# was also a good choice.

Steamworks, which is provided for free, offers real-time sales data and bug-reporting statistics; official groups, through which we can communicate to customers; creation of in-game downloadable content; anti-cheat protection; and many more helpful features to help us expand gameplay while safeguarding the product. Incorporating Steamworks also allows customer to always have the latest version of the product, so redistributing is not an issue if something changes in the code. As Owen Godfrey puts it in his thesis, “*Full Steam Ahead*”, “it merely needs to be connected to the internet to download any software updates…ensures that when patches are released they are applied to the game without the need of player interaction.” (pg. 36).

According to the thesis, “*Testing and Maintenance in the Video Game Industry Today*”, written by Anthony Jarman, although one usually thinks of bug-fixing as the most important part of software maintenance, in reality, it takes up a relatively small amount of resources, because most of the necessary corrections are found and fixed fairly early after the software’s release. Anthony Jarman also states that, “…the most common type of maintenance used on video games is perfective maintenance.” (pg. 21) and, “In the video game industry today, the most common use of perfective maintenance lies in the area of downloadable content.” (pg. 21). The DLC can be sold to customers and provide more money for the company to hire other developers, if necessary.

Even after the pricing and releasing of the game, feedback from users will still be very important, so regular checks of the company email account, Facebook business account, and Steam forums will be essential as they identify more specific faults in the product or make suggestions for improvements. The first step to solving a customer-identified problem is being able to recreate the problem. This will require more test suites to ensure that the problem is taken care of without introducing other problems at the same time.

Also important to keeping customers interested is providing news about potential, new developments or content to show them that we are still interested in the game ourselves and are listening to their suggestions. Sequels, re-releases, and expansion packs can be great, not only for the profits they produce on their own, but also for renewing interest in the original game and after the original $100 fee from Steam, our group could post any number of new games to attempt to get a green light on with no additional charge.

Eventually, an additional code developer may be needed to work on the project since we, the originators, are still going to school. If the group decides to hire an intern-level developer, perhaps even from the university, that person may be willing to work for $14-$20 per hour. A professional developer might need $30-$40 per hour, but they would be able to devote all their time to the project. Hopefully not, but if the involvement of a lawyer becomes necessary because of piracy or any other criminal charge, a normal fee, according to the article, “*How, and How Much, Do Lawyers Charge?*” would be around $200-$300 per hour plus expenses.

Meanwhile, we will also continue to try and find new ways to advertise our product to new markets. If the game becomes more popular, it will be easier to find people to review the game or showcase it on a YouTube video. The game’s presence on Facebook should also be increasing during this time. Based on Jack Marshall’s informal data-gathering in the article, “What Online Ads Really Cost” we could advertise on a $30 cost-per-thousand views basis on Hulu, so even if we end up pricing the game at $1, the returns of advertising like this could be very good. Different promotions, such as free trials for a weekend, would also generate more traffic. Demonstrations of the game could be distributed to different free-to-play game websites with links to the store where the full version is sold.

As difficult as it may be to get to that final release point on the Steam store, this service will effectively “…remove the costs of production and also cover the costs of distribution…” (pg. 16) as Godfrey puts it in his thesis. Even if the group decided to quit maintaining the game altogether, we could still earn some profit simply having Wizard Needs Matches sitting on the “shelves” of Steam, making it a good investment.

Jarman, A 2010, Testing and Maintenance in the Video Game Industry Today. Computer Science Senior Honours, Liberty University.

Godfrey, Owen. *Full Steam Ahead*. Thesis. University of Wollongong, 2012. N.p.: n.p., n.d. *Google Scholar*. Web. 14 Dec. 2016.

North Carolina State University. "Software Maintenance." *Realsearch*. Realsearch, n.d. Web. 14 Dec. 2016. <agile.csc.ncsu.edu/SEMaterials/MaintenanceRefactoring.pdf>.

Valve. "STEAMWORKS." *STEAMWORKS*. Valve, 2011. Web. 15 Dec. 2016. <http://www.steampowered.com/steamworks/index.php>.

Marshall, Jack. "What Online Ads Really Cost." *Digiday*. N.p., 22 Feb. 2013. Web. 15 Dec. 2016.

Anonymous. "How, and How Much, Do Lawyers Charge?" *Lawyers.com*. Internet Brands Inc., 09 Apr. 2015. Web. 15 Dec. 2016.