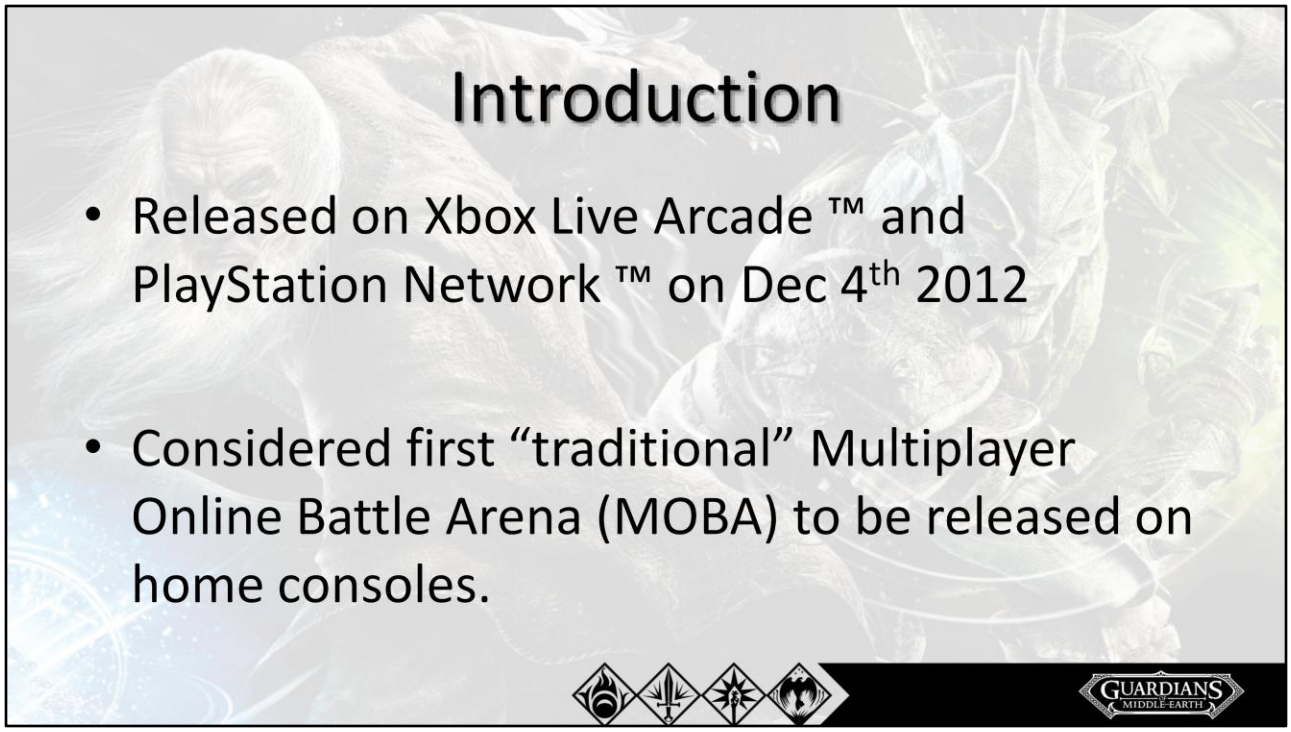
Hi, welcome to my talk. Guardians of Middle-earth: Adapting the MOBA for the Console Crowd
My name is Evan Lewis, a UI Engineer at Monolith Productions in Kirkland Washington.



The team at Monolith wanted to take the exciting world of Tolkein's Lord of the Ring's series and allow players to face off against each other as their favorite characters in thrilling and tactical battles. This made the MOBA an obvious choice.

Uber Entertainment released Monday Night Combat in 2010, and Ronimo Games launched Awesomenauts in 2012. These two could definitely be considered in the MOBA genre, but they are more adaptations into the 3rd person shooter and 2D platformer genres respectively.

Thus, you could consider GoME the first "traditional" MOBA to be released on home consoles.

How did we do it?

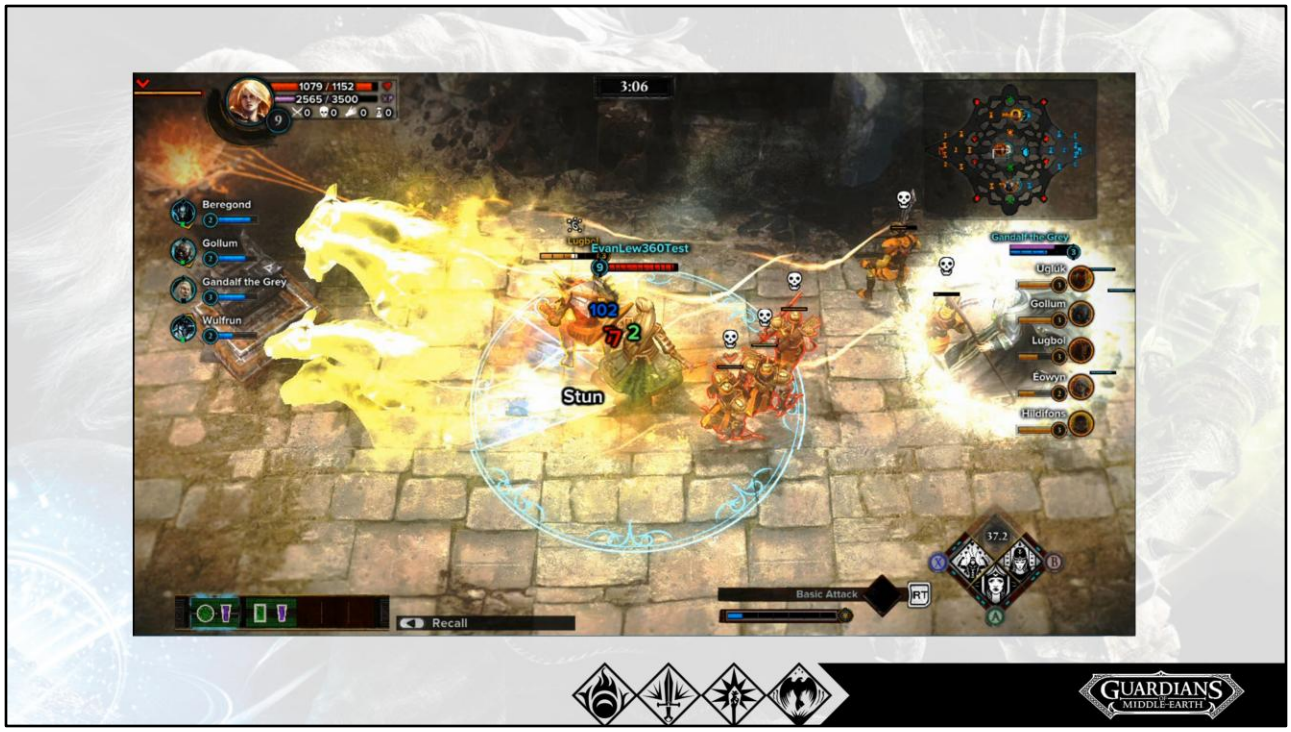
- Scaleform 4.0.16
- Actionscript 2.0
- Insanity

This photo is of the rest of my UI team on GoME. (from L to R)
Top Row:
Ted Snook – SeniorUI Engineer
Marshall Beachy – UI Engineer
Evan Lewis – UI Engineer
Stephen Whetstine – SeniorUI Artist/Designer

Bottom Row:
Tyler Carpenter – UX Designer

We used Scaleform's flash implementation alongside Actionscript 2.0 to make our UI systems in order to showcase all the awesome in the game.

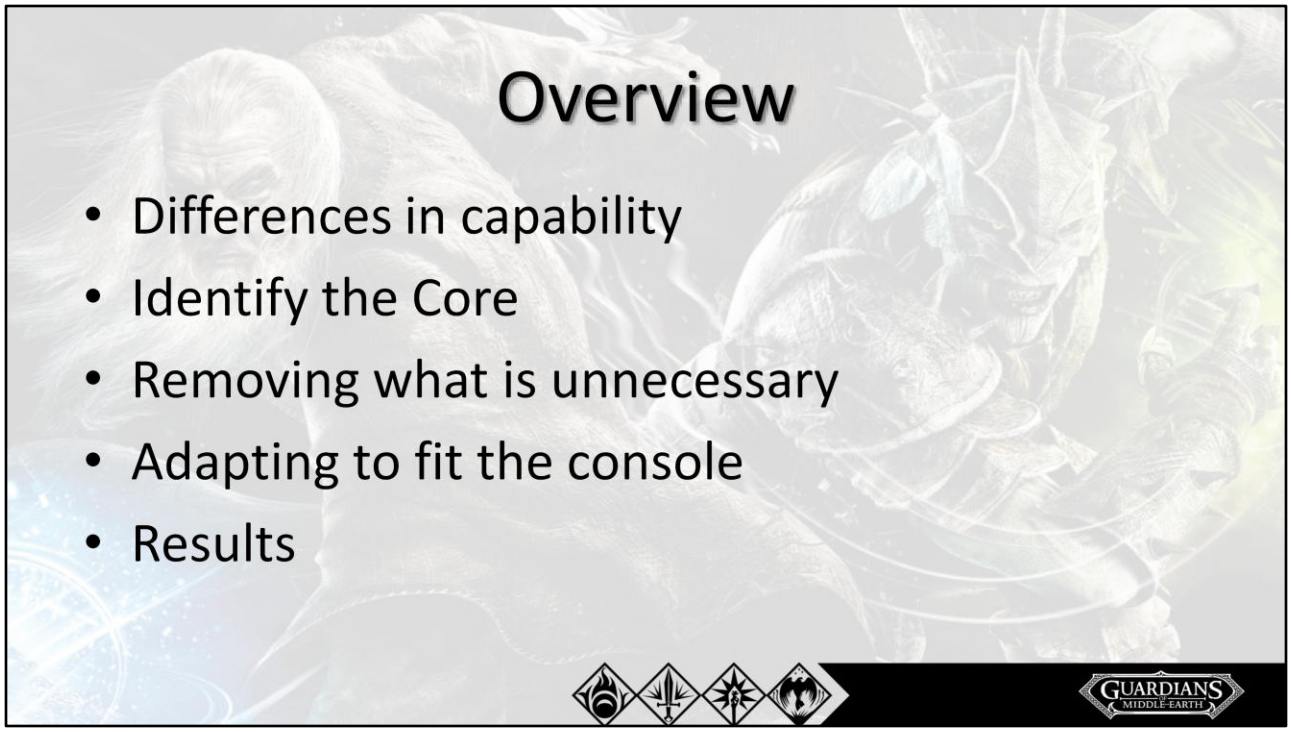
Basically I just wanted this photo to make people laugh and to thank the rest of my team who couldn't make it to GDC this year.



This video showcases some excitement of combat in GoME from a close up perspective. The user can be seen firing off a few different abilities and seeing all the damage numbers and UI indicators associated with them.

It also gives a brief glimpse of some of our in game menu.

There is a lot of data to be seen here to be sure, and this talk should explain how we managed to make it all work.



Overview of what this talk will cover.

First I am going to describe some of the inherent Differences in Capability between the PC and home console systems. This includes the hardware limitations placed on us and the different play styles and habits of players between the two systems.

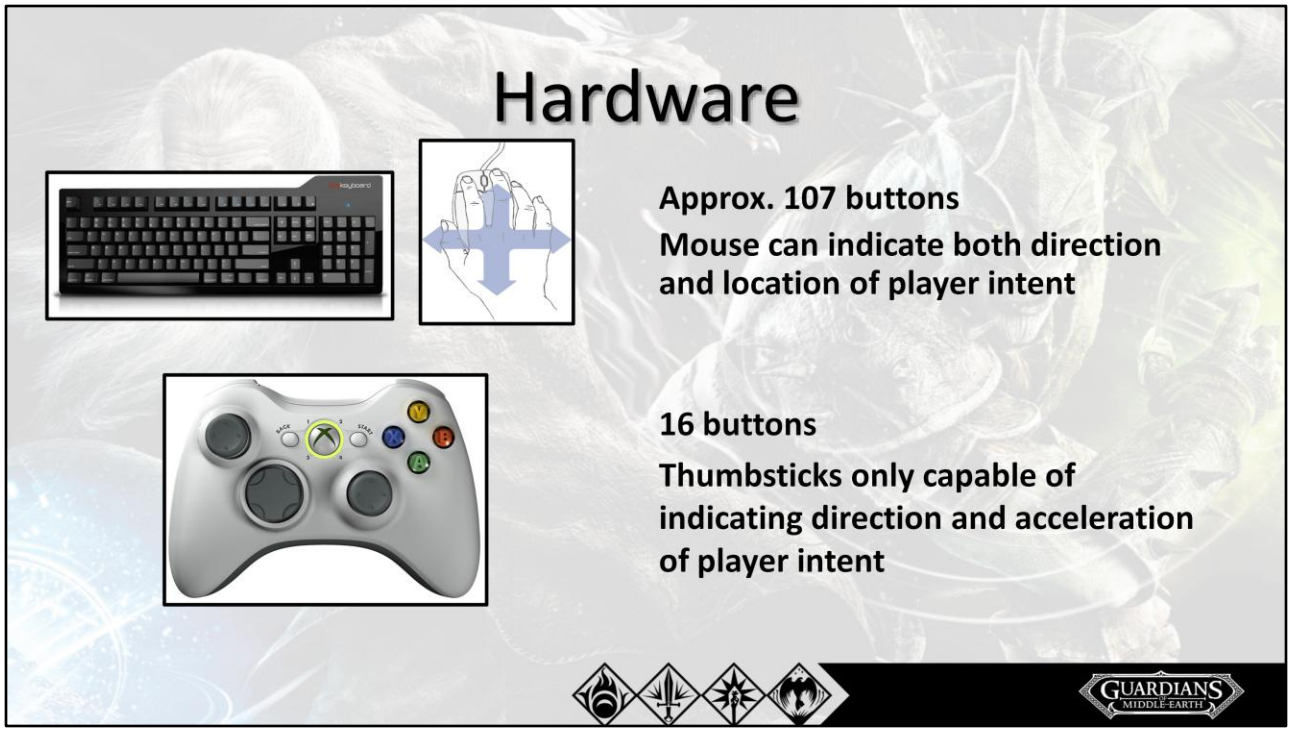
A key part of the entire project's design step was to Identify the Core of the MOBA experience and acknowledging the expectations of players of the genre.

At times it was necessary to Remove parts from the game in order to make the adaptation and we will go over which parts were cut.

And which parts were kept as I go over what we changed and Adapted to the console and how that process came about.

And then a brief overview of our Results and how everything turned out.

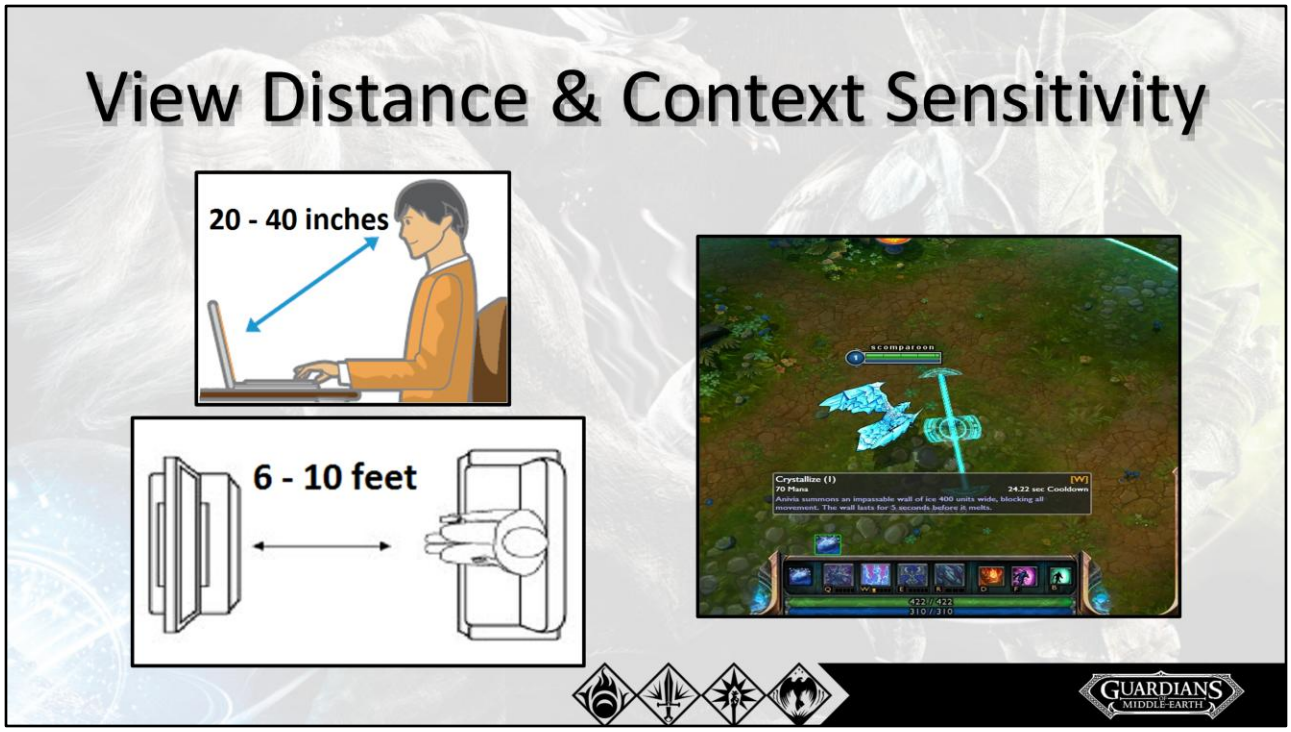
Shifting from a PC to a Console has a great number of limitations built in from a User Interface perspective



The most obvious : Hardware.

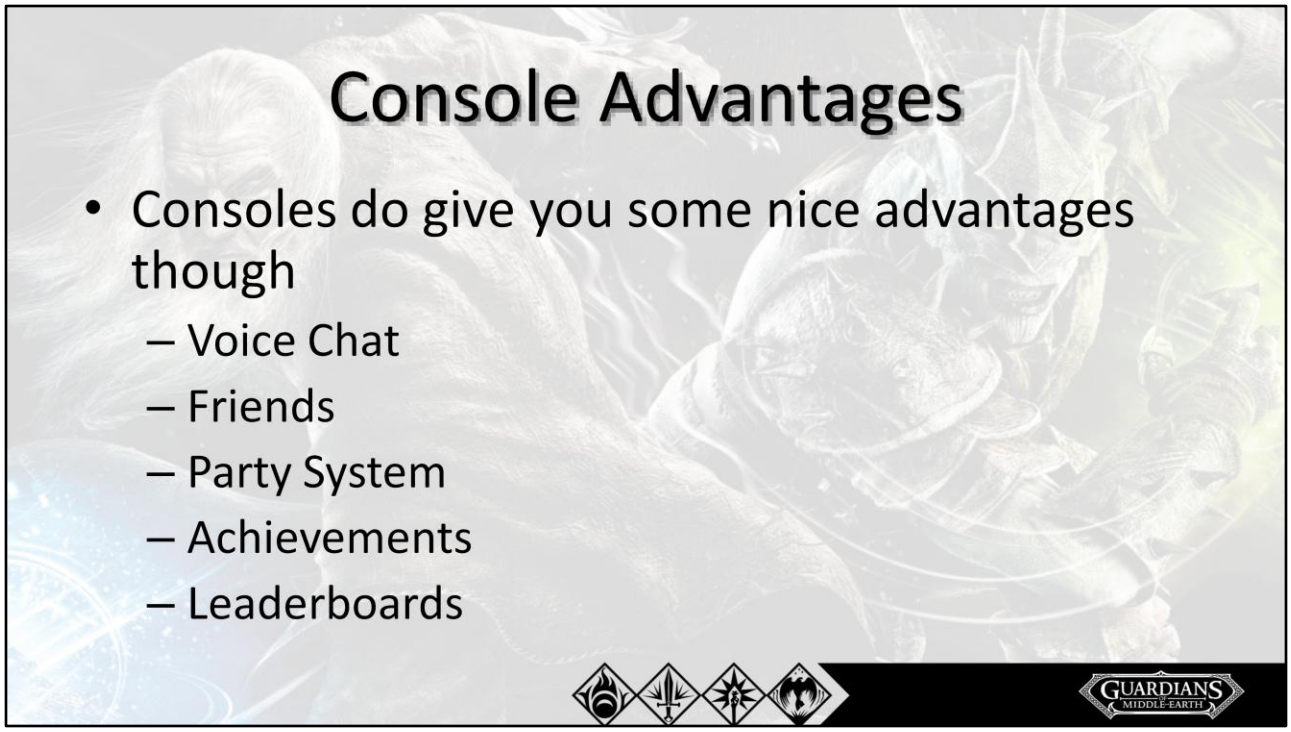
The most obvious and noteworthy is the difference in control method. A standard keyboard has approximately 107 buttons, and a mouse that can indicate player intent, as well as location as the player can create an a priori perception and muscle memory of the location of their hand on their desk, and the location of a cursor on their screen.

A console controller, like the Xbox 360 one shown here, has only 16 buttons. While it has dual thumbsticks, they can only be used for the direction and acceleration of the player's intent. This means that the number of possible actions available to the player must be simplified, or increased via button combinations.

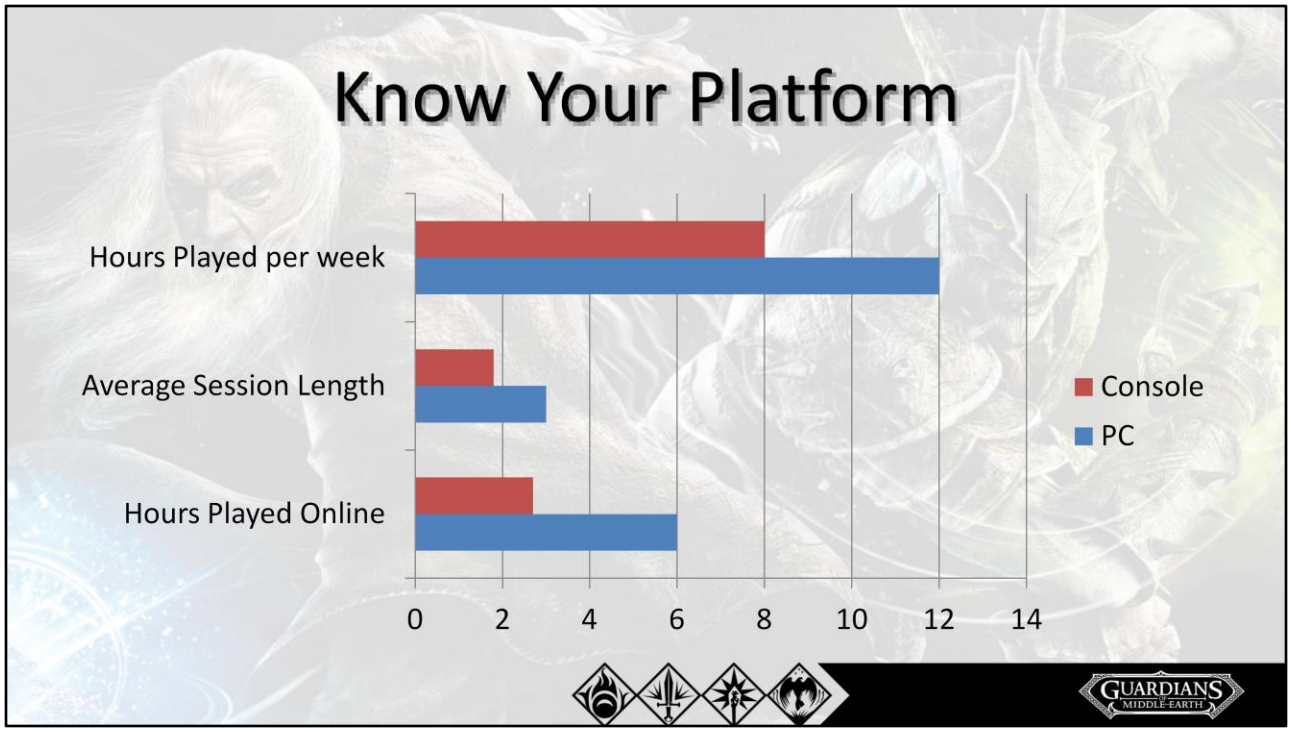


A typical person viewing their PC monitor sits about 20-40 inches away, while a person playing a game on their television tends to sit 6-10 feet away from their screen. This results in visibility issues, especially apparent in the MOBA as they tend to use intricate icons to indicate various information.

As mentioned before, without a cursor on the screen that one has from a mouse, hover-over or context sensitive tool tips are far more difficult. In PC MOBAs these are used to give the player vast amounts of useful information



Console development does give us a few great things built into the system. Voice chat, friends lists, and a party system are essential for community and teamwork based games like MOBAs as players have to coordinate tactics in game and matches outside of the game with each other. Achievements and Leaderboards are really great for helping to build a thriving community around your game which is really necessary for MOBAs to succeed. Having a pre-made, solid back end available for these features is a real boon when developing a MOBA.



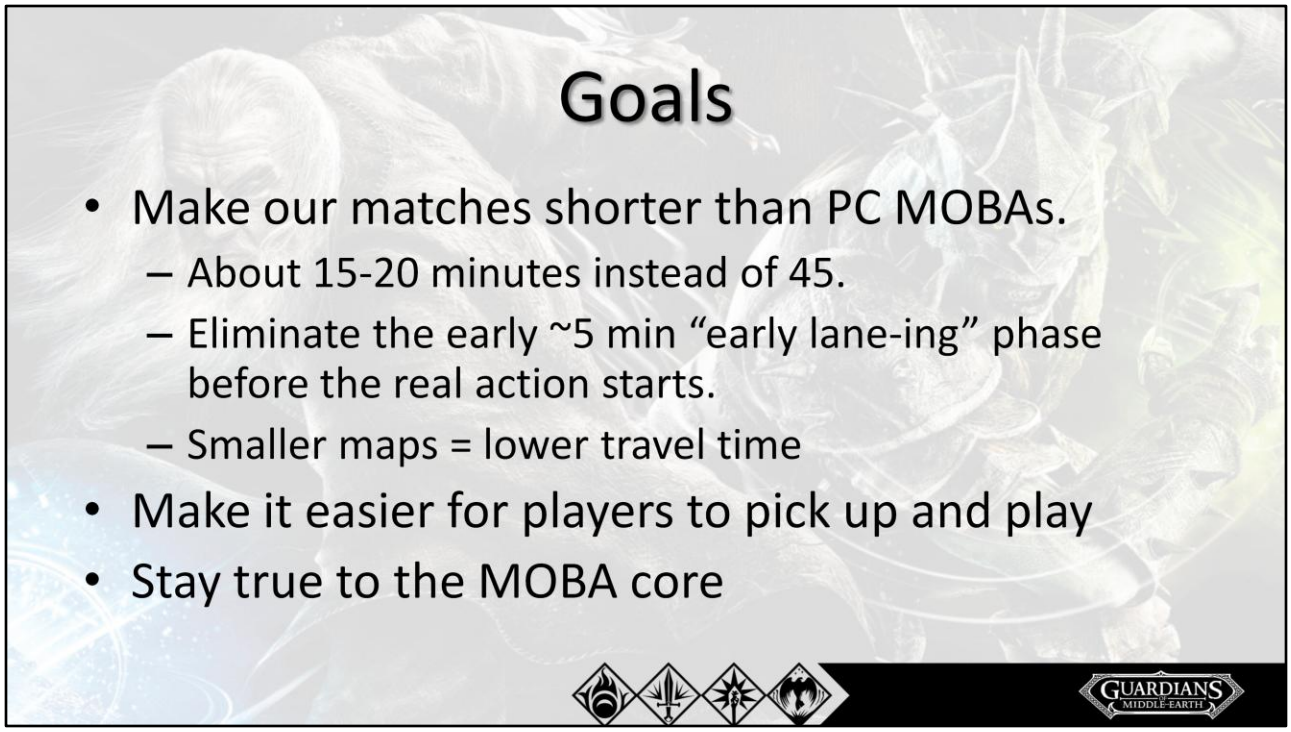
It’s also important to recognize some of the key demographic differences between PC and console gamers.

Avid PC gamers tend to spend an average of 12 hours a week gaming on their PC while console-primary avid gamers trend closer to 8 hours.

The average length of a console gamer's play sessions tend to be 2/3rds to 1/2 the length of a PC player.

Console players tend to play 1/3rd of their gaming time online, compared to the 50% of the time of PC Gamers.

These sorts of differences showed us that we would have to bend our experience towards a shorter engagement time, but console players are still hard core and we wouldn't win any points "dumbing down" the MOBA for them.



Based on this information some of our goals and strategies for development became.

Make our matches shorter.
We wanted a time frame of around 15-20 minutes to be more in line with typical player engagement times. Thus there was also a desire to get right into the thick of the action so we wanted to remove the early lane-ing phase. To help with this we made our maps about 60% the size of other MOBAs to facilitate players respawning and getting back into the action quicker.

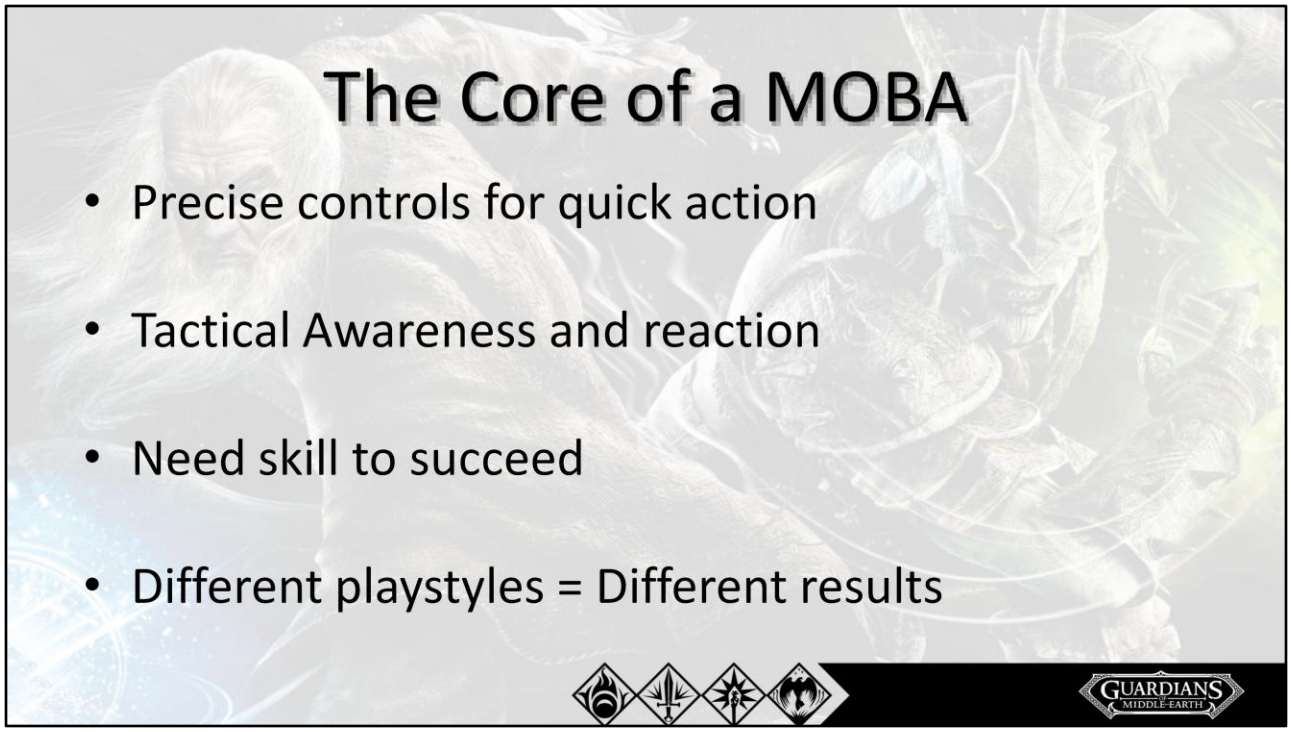
Make it easier for players to just pick up and play so that the game would be both more accessible, and easier to play one quick match before work, or before dinner.

Most important: Stay True to the MOBA core. The MOBA community is big and they treat MOBAs as serious business. They won't tolerate anything that strays too far from the core of the genre.

We need to identify what the core of a MOBA is, so we can keep as much of it intact as we make our conversion.

Thus we absolutely could not approach the concept this way. Despite how amusing Yahtzee's comic may be, we had to respect both groups equally and find the awesome medium that would appeal to both. We had to figure out the central nugget of MOBA awesome that was central to the genre.

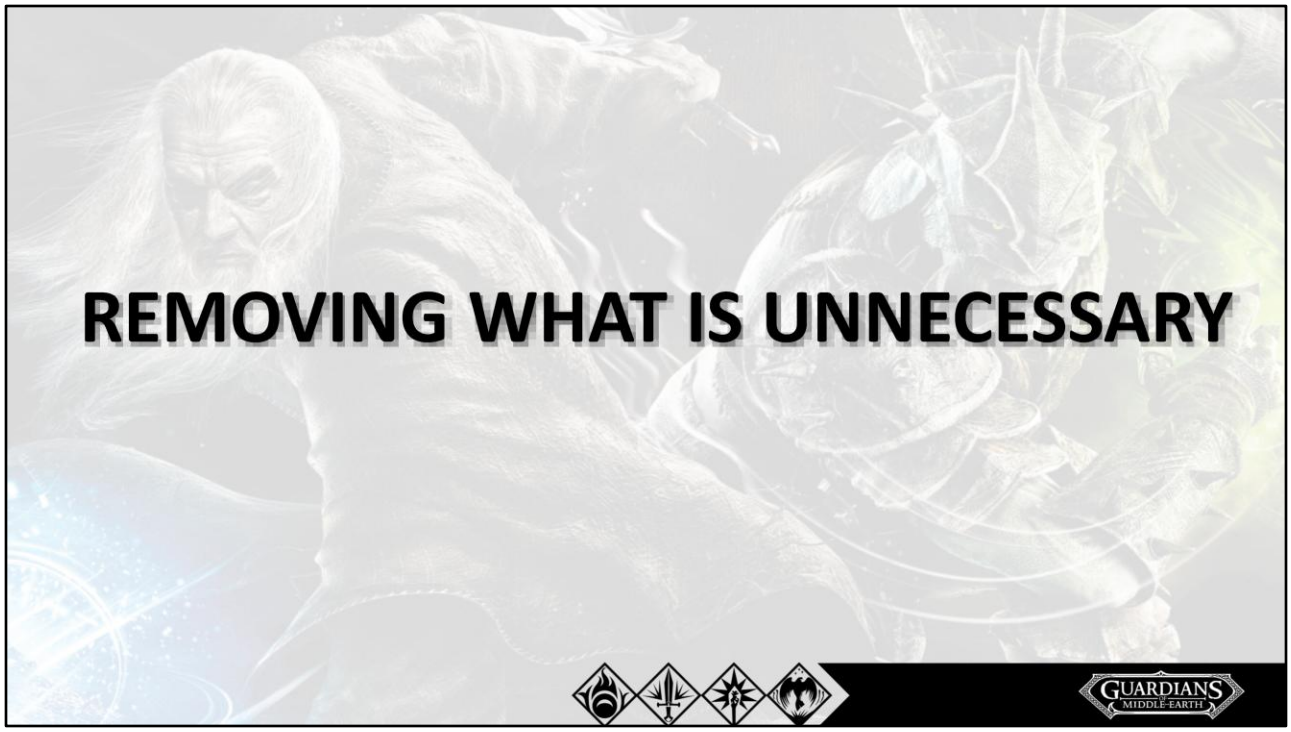
For instance an FPS could have a core centered on 3D aiming, guns, a targeting reticule, health, and moving and strafing.



For a MOBA that core would be:

- The ability to rapidly change direction in response to threats.
- Being able to see the state of the battlefield and form tactical decisions based off that information
- The ability to precisely direct abilities so that skill is required to maximize their effect;
- Changing your hero's base stats so that two people can face off with the same character, yet not exactly identical.

With these core concepts in mind, we set out to make our game. During development of our controls and UI we would often have to ask "is this core to a MOBA" and compare it to our criteria, and ask other die-hard fans of the genre, if the response was something that was "meh, its useful but not super important" we knew we were safe to tweak or remove it.



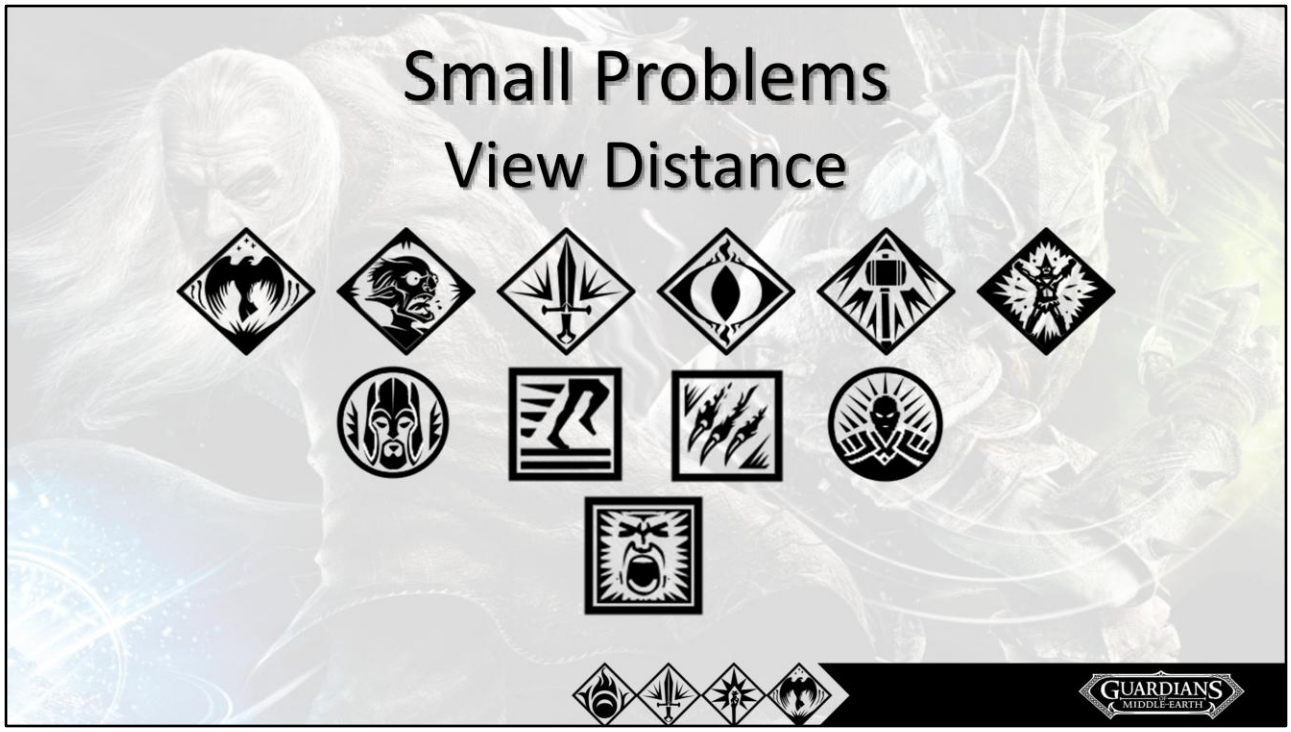
With the limitations and different expectations expressed before, we knew that aspects of the standard PC MOBA would have to get the axe.

We eliminated the in game store.
While it is a source of great game complexity, allowing player to adjust strategies on the fly, we wanted to keep game times shorter ,thus didn't want players spending loads of time comparing items to buy. We opted for a loadout based system that was edited between matches when players had plenty of time to compare and consider their choices without it affecting their team's success.

Last-hitting.
Because we removed the in-game store last-hitting wasn't needed for a currency, also directly targeting the one enemy out of the group around you for a precise strike ended up being troublesome and more game-delaying than anything and would often lead to kill-stealing. We traded that out for simply dividing XP amongst the damage dealers and anyone nearby in the lane.

Mana.
It was apparent pretty early that "mana" served primarily as a way to limit characters in early game, but became less of an issue in late game conflcts. Given our shorter game time and desire to get right to the action, we opted to remove mana and made cooldowns the only real resource management system.

We had to closely examine each aspect of the remaining systems and figure out what challenges remained in adapting them the new setting and how best to make the conversion.



As the player would be sitting 10 feet away from the screen we had to make some adjustments for visual fidelity. First off our icons. We were unable to display the fancy multicolored and overly intricate designs that one might usually find in a MOBA. Instead, we opted for a simpler Black and white approach to our icons with large shapes that were easily recognized. Furthermore, we shaped the icon overall according to what type of effect they were. Be they diamond for Guardian Ability, circle for Potion, or square for Command.
This allowed players to quickly ascertain what icon tied to what in game action.

Small Problems

We learned via frequent playtests that recognizing exactly which enemy is attacking you could be a bit difficult, thus we added a red chevron on top of enemy health indicators to let you know exactly which soldiers have you targeted. we left these off of player-controlled entities as deception and leading players on is a powerful tactic in skilled competition.

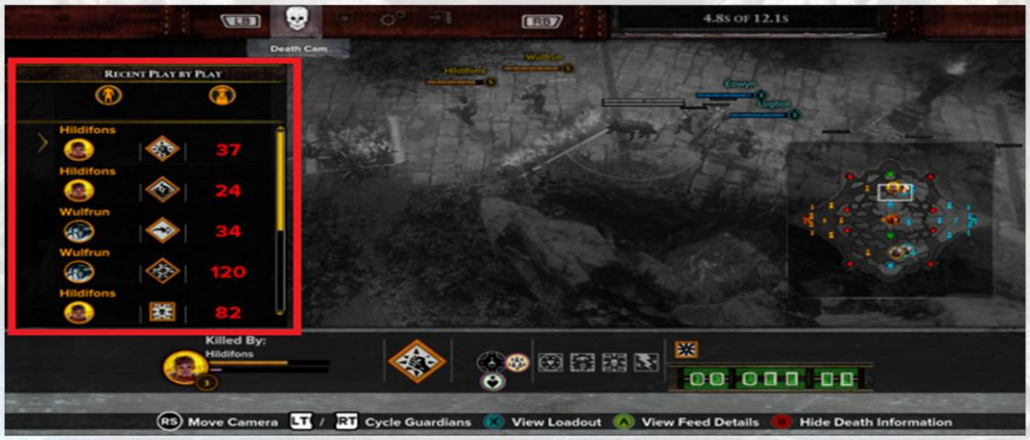
Lack of mouse-over ended up not being quite the difficulty as originally expected even though most information about skills, buffs, and other potentially game-changing data is often displayed via these tooltips on PC. Thus, we had to arrange for different ways to display a lot of this information.

Players need to know what skills do what when leveling up. Thus by holding RB, they are able to quickly bring up a window that shows them a brief description of their skills with numerical values inserted and release the button to hide it just as quickly. This is also the screen used to level up in game so players can choose to read closely if they want or simply pop it open quickly to spend a skill point. We also took the opportunity to include the player's unique Guardian Abilty while this screen was open to act as a reminder to the player. Since this screen is only open while RB is held down, and typically not for long, we were able to devote more screen space to explaining skills and their effects.

Small Problems

In order for players to understand what was happening to them at any moment, we placed the indicators for what status effects the player was experiencing near their main health bar on the upper left hand side of the screen. Their recognizability was greatly helped by the previously shown cardinal shapes and icon design. Playtest feedback also indicated that players desired a more emphatic indication of when someone was affected by a Command. The Commands are tremendously powerful abilities that have long cooldowns, and often short durations. These can include massive speed or health buffs, or stunning and blinding opponents, thus is was very important to know if an ally or foe was under its influence. We decided to place these indicators on the player's moving health bar to make it more obvious when one of them was in effect. Since the duration of most of these effects was so short, we felt comfortable with their position not obscuring the character for long.

Small Problems



In the chaos of battle it can often times be confusing to a player as to the exact cause of death. We initially considered doing a death replay system, but soon decided against it as it would disrupt a players engagement to the ongoing conflict. Thus we added a feed to the screen while players waited to respawn. This scrollable information display showed the most recent effects to befall the player and their relative health cost. This allowed our users to discover what series of events lead to their demise so they could learn from their mistakes. This proved really useful for new players to learn what abilities other characters had as well during the relatively inactive time of death.

Loadout Visibility

Knowing what equipment a foe brought into a match was difficult. Originally on our pause menu we showed the minimap alongside the current score and status of the players.

We allowed a player to see what enemy loadouts were via selecting another player…

Loadout Visibility

… choosing to view their loadout…

Loadout Visibility

… and bringing them to this information screen.

Feedback from our expert players indicated that even with the wide variety of loadout options, experienced players would be able to identify all the parts of another's loadout simply by a glance at the icons associated with them.

Loadout Visibility

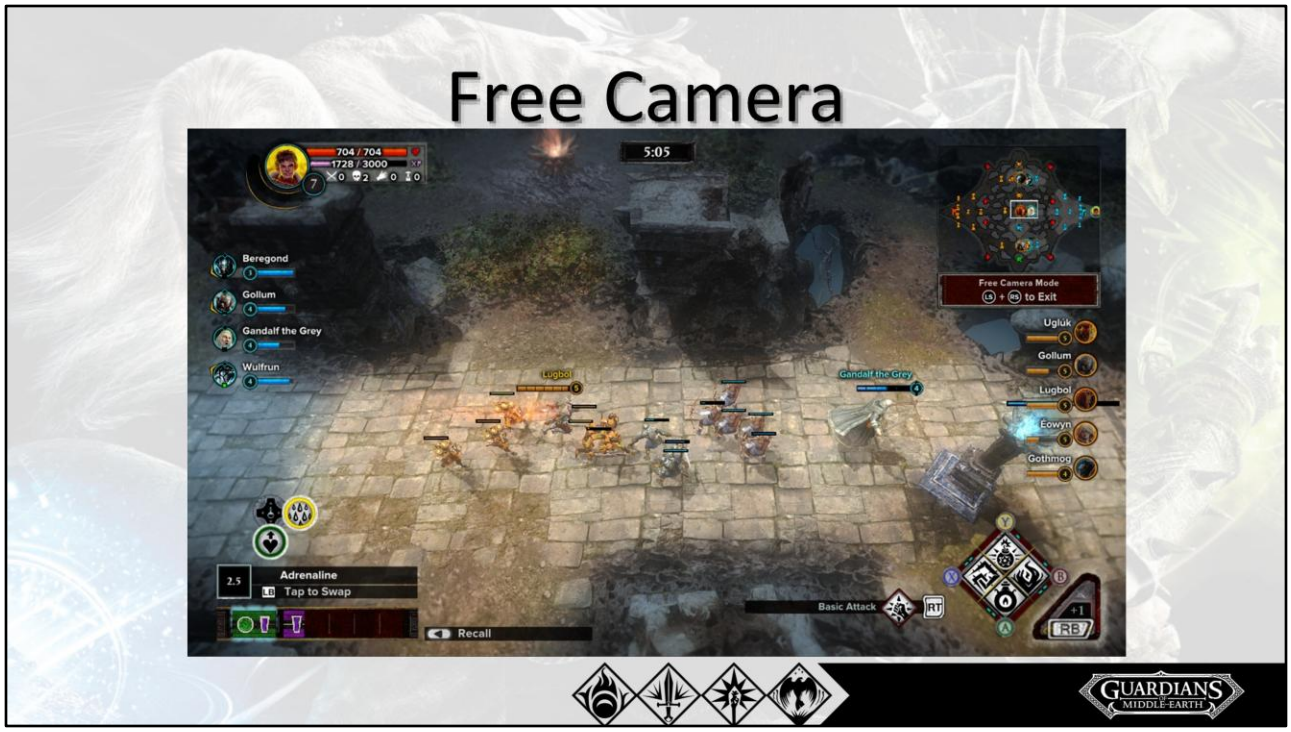
Thus we decided to speed access to this information by replacing the minimap with a loadout preview of the opponent currently highlighted. This was preferred as, while you still had to bring up the in-game menu in order to access this screen, it was easy to quickly hop from player to player and get a read on what they were bringing to the fight.

Finding this sort of balance between ease of access and breadth of information was one of the larger and more fun challenge of designing this User Interface.

Character Attributes

Players tend to enjoy trying out various different combinations of loadout options and comparing them against each other. Truly die hard players liked to place numerical values on these differences in order to more efficiently min max the system. Initially we had requests to place data such as current attack power, and regen rate on the screen at all times.

We realized that it was only those die hard players who really wanted to put numbers to their character's success, and that that much data on screen always looked pretty bad. Thus, we realized we could put this information in the same place as the full loadout display, behind two clicks on the in-game menu. The data remained available, just unobtrusive to players who didn't really care about exact numbers, so long as they kicked more butt.



A real strength of PC MOBAs is the ability to click anywhere on the minimap and have the camera move to that point. Unfortunately we were unable to make a possibility without a floating cursor. Initially we brushed this off as unnecessary given that our maps were about 60% the size of other maps to facilitate a faster game. However this was highlighted as more of a core feature, the ability to monitor other parts of the battlefield to advise or castigate a teammate. Thus we adde a free-camera option under the control of the Right Stick which was activated by Double-thumbclick. This feature was also turned on by default when a player was dead and waiting to respawn to give them something to do as they watched the respawn timer count down.

I feel I should note how important this feedback was. We brought in several professional MOBA players to try out the game and give us feedback (you should check out the behind the scenes videos on Monolith's youtube channel), and we also assembled a Play-balance team composed of giant MOBA fans. Many times the dev team would think some information was necessary only to have our die-hard MOBA players tell us "sure its there, but no one uses it. its not a core feature" allowing us to hide that information deeper in the menus. These resources and nearly daily playtests with feedback from everyone about the UI was a invaluable to us.

Targeting v1.0

With the simple problems solved we come to the big one: Targeting control and feedback.

We were quickly able to realize that placing a cursor on screen and using click to move and attack just like a PC wouldn't work very well, so we changed things up. Making the left thumbstick in charge of player motion seemed an obvious choice and players responded well to it... but we pretty much left everything else alone. In our initial phase the player moved a cursor around the screen with the Right stick, and would click with Right Trigger to attack enemies or activate abilities if they had been primed with a face button. This was mostly a disaster. While players liked the face button association with abilities, they would often lose track of the cursor if a battle got heated, and would get frustrated when attacking someone outside of the range indicator circle would result in the character moving closer to attack. Also, spamming RT to attack was tiring.

Moving on we decided to simplify things a bit. Since we were controlling the player motion with the left stick, we no longer had to give the player the option of 'clicking' on an opponent across the screen to move and hit them. Thus, we restricted the targeting reticle to be inside the maximum range of the chosen attack. This way a player would always be able to see who they could attack at a glance. The problem this approach had was that players would still lose track of the cursor and direct targeting was still an issue as you would often have small clusters of enemies.

This is when we thought of our first breakthrough 'What if we remove the 'click to attack' idea and just make the player attack as long as the button is held down, and there is a valid target.' Initially there was some push back about how could you have a MOBA without clicking? They go together so well, how else will we rate APM? We realized, however, that the core issue was that the player should have a fairly easy time targeting who they want and that is more important than the simple joy of clicking a bunch.

Targeting Feedback

These concepts were effective and remained in use for awhile, but now we had a greater problem, how do we indicate the chosen target? We went with a system that utilized multiple levels of redundant feedback.

To begin we traced all enemies to make it easier to distinguish them from the environment

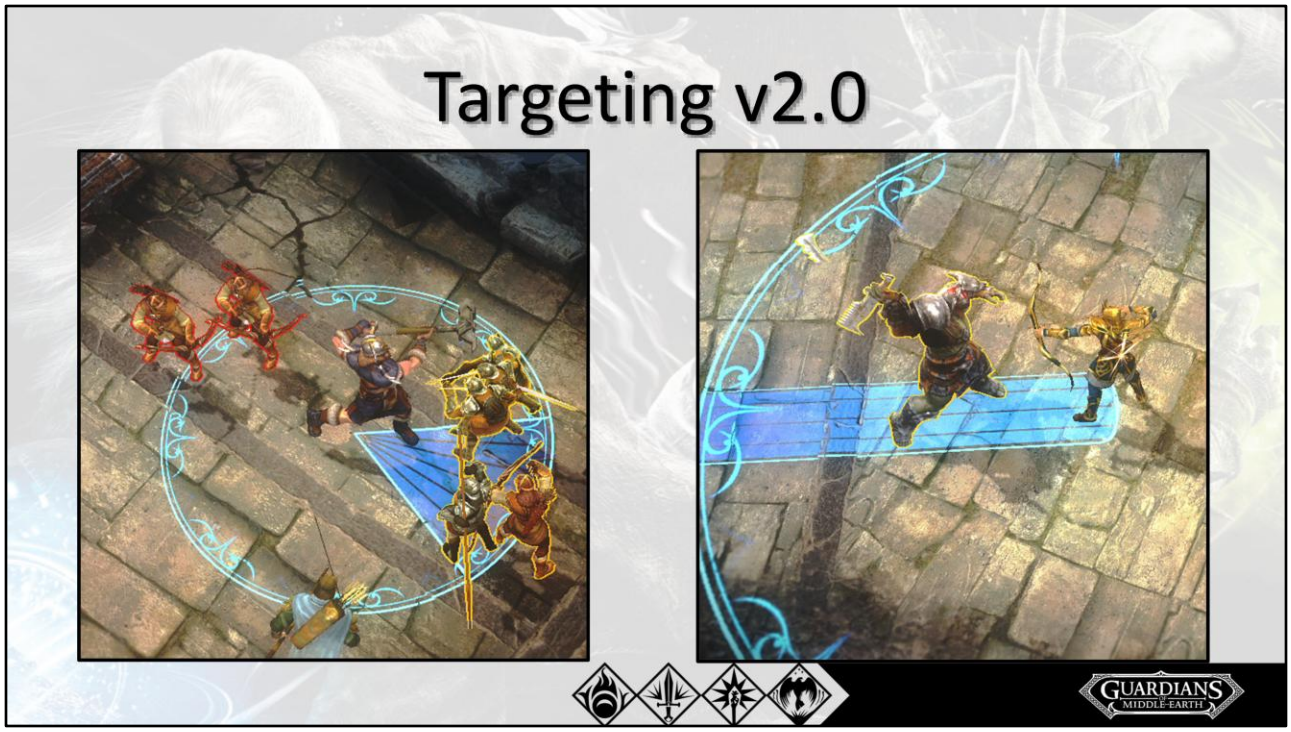
Targeting Feedback

Given that we had a moving cursor that the player would focus on most of the time, we changed the color of the cursor to match friend or foe colors of Orange and Blue in order to link the two together in the players mind.

Targeting Feedback

Given that the player had a range ring around them at all times we also adjusted its color to match the current status of a target to help solidify the relationship and help call out AoE attacks. We noticed that players were flocking to characters with centered AoE attacks during playtests as they were remarkably easy to target and felt really satisfying to the user in this setup.

It was then that we had our eureka moment. 'Make it as easy as possible for the players to hit who they want, even if it means hitting some extra guys' “ in other words ‘AoE is king!’



So we made every ability be AoE in design, players simply had to make sure the desired target was in a general area before firing off their abilities. But what to do about Basic Attacks?

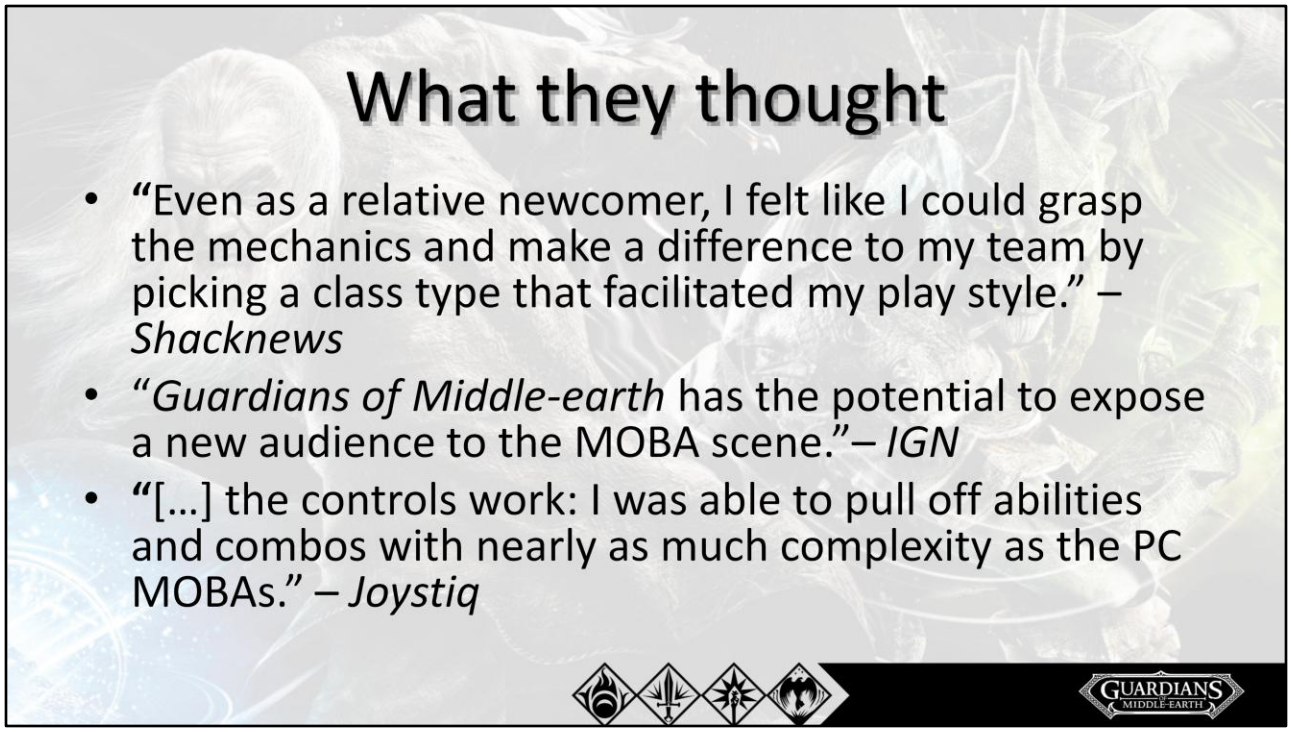
We arrived at the targeting wedge or line. We made the basic attacks of melee characters large wedges which that would simply need to hold down RT and they could hit all the enemies within that area. Ranged characters had a similar target area - a long bar. This meant melee characters, though handicapped in strike distance, could easily dispatch large numbers of nearby enemies. Ranged characters couldn't hit as many when surrounded, but could choose angles of attack that made them deadly from the right positions.

Targeting v2.Awesome

By applying this style of AoE to all attacks, it became easy for players to target what they wanted as they just had to make sure the foe was in that large area. Players no longer had to keep track of the players position and a separate cursor as the AoE was attached the player so they could learn its location relative to where they moved the player easily.

That's the process. Guardians of Middle-earth turned out to be a pretty great game that I and the rest of the team at Monolith are really proud of.

Machinima liked us and gave us Best Downloadable Game of E3 2012



And most reviews and player feedback praised our controls and UI for being accessible yet profound.

The most telling thing for me was how often our team was stopped at events where we demoed the game and being asked the same question "How did you manage to make a PC genre like the MOBA feel so at home on a console controller and tv?"

And this is my incredibly intricate response.

Please feel free to contact me at Evan.Lewis@lith.com for any questions or comments on this talk or on GoME in general.

References

- Extra screenshots used from League of Legends ©Riot Games
- Gamer demographic information gleaned from NPD Gamer Segmentation Report 2011