Tourney Journey - Design and Implementation

CS4303 - Video Games

My design document went through huge changes as my game was being developed. I had originally planned to create a top down based game. The player would still fight through waves of enemies and progress through levels, but fighting would be done with different equipment which would drop from enemies, not by jumping on them. Furthermore, there is no mention of a torus, slowing down time, or gravity in my original design plan.

The reason for these discrepancies is that many of the ideas which ended up in my final product occurred to me as I was developing the game. I actually started out developing the game as a top down game, then decided to make the maps into toruses. I then had the idea to make the player fall downwards at a constant speed when a button was pressed. It was at this point that I realized that it could also be a top-down game, depending on perspective. I then started developing the physics for this, and eventually ended up discarding the top-down idea altogether.

Despite these changes, however, there are still elements of my original idea present in my game. The wave/level system discussed earlier is one example. Additionally the game still combines the idea of a Tourney where the player progresses through a series of increasingly difficult fights with the idea of a Journey where the player also covers geographical distance as the game progresses.

Ultimately, I feel that this ad-hoc approach to game development works better than specifying a design document at the start and then sticking to this no matter what. My original specification was not able to adjust itself in response to how the game was evolving, whereas I would argue this less rigid approach resulted in a more organic game than it would have been had I stayed with my original specification. The original specification was useful in that it provided a starting point which I could build from, but it did not account for hindsight, which means that I had no way of gauging exactly how difficult the project would be when I embarked on it. This real-time approach instead allowed me to make adjustments to my approach in real time both in response to the time frame I had to develop the game, as well as non-related time commitments which came up unexpectedly, which is why this approach was the superior one to take.