1. Please provide an update on implementation.

Greetings Professor Cassens, I once more can report that the implementation of DungeonBuddy remains on-schedule for the time being, though the unexpected focus on Optimization will delay the onset of full-fledged testing until next week. It was definitely worth the 5 hours or so spent so far, however.

**Week 4 (Nov 4 – Nov 10)**  
Week 4 was all about getting NPC functionality hooked up across the entire program, focusing on the NPC Tool (A random NPC Generator), and the NPC Tracker so DM’s can keep track of and edit manually or randomly generated NPCs in their game. This process surprisingly didn’t take very long, and I found myself in a sort of yolo mindset at the time so I went ahead and proceeded on to one of my stretch goals, the Magic Item Shop generator. Programmatically the Shop generator drew components from the Random NPC Generator and Encounter Tool pages, which cut down on the impromptu workload significantly. There’s a few minor tweaks I’d like to do to both generators eventually for stylistic purposes (give Magic Items colors based on item rarity), but they are both completely functional.

Also… The Website is up and running! <https://dungeonbuddy.net/>

*Total Time Spent*: 14 hours 30 minutes. Not counting extensive Server Host setup.

**Week 5 (Nov 11 – Nov 17)**

Week 5 is the week of the Custom Page Tool, which is to allow DM’s to upload PDF documents to the server so that their Players can access them. This was half finished when Optimization suddenly came out of nowhere, and was subsequently wrecked by changes made to the overall website (use of a master page with virtually every page), but it has subsequently been repaired. The core functionality of the tool (Add/Delete/Rename/Reorder) is complete, with just a couple of easy fringe case bugs remaining to be addressed (prevent duplicate database entries from occurring type stuff). I expect to begin work on the Game Information tomorrow, and complete it by end of week Sunday on schedule as it doesn’t involve anything too complex.

*Total Time Spent so far:* 9 hours 35 minutes

2. Provide a breakdown of your contribution if you are in a team.

I am the best team member I could have ever asked for, and currently carry the team.

3. Provide previous and current timelines.  Describe any changes and why they were made. Be specific on your timelines.

**Previous Timeline**

Week 1 (Oct 14 – Oct 20)

* Reach Design decision on use of Grid View, ListView, or my own Object Table, for the many tables I need.

Week 2 (Oct 21 – Oct 27)

* Login/New User registration completion
* Encounter Tool completion

Week 3 (Oct 28 – Nov 3)

* NPC Tool completion
* NPC Tracker completion

Week 4 (Nov 4 – Nov 10)

* ‘Create Game’ Page completion
* ‘Join Game’ Page completion
* ‘Game Party’ Page completion

Week 5 (Nov 11 – Nov 17)

* Custom Game pages completion
* Create Game Information Page
* Begin Project Testing

Week 6 (Nov 18 – Nov 24)

* Finalize any remaining functionality of Main Project
* Begin work on secondary project (Desktop app)

Week 7 (Nov 25 – Dec 1)

* Complete login/authentication feature of Desktop application
* Complete Desktop application’s ‘Import/Export’ functionality

Week 8 (Dec 2 – Dec 8)

* Present Project Demo and Testing
* Testing Complete
* Working Demo Complete

**Current Timeline**

Week 1 (Oct 14 – Oct 20)

* Reach Design decision on use of Grid View, ListView, or my own Object Table, for the many tables I need.

Week 2 (Oct 21 – Oct 27)

* Login/New User registration completion
* Encounter Tool completion

Week 3 (Oct 28 – Nov 3)

* ‘Create Game’ Page completion
* ‘Join Game’ Page completion
* ‘Game Party’ Page completion

Week 4 (Nov 4 – Nov 10)

* NPC Tool completion
* NPC Tracker completion

Week 5 (Nov 11 – Nov 17)

* Finish Custom Game pages
  + Fix fringe issues like duplicate CustomPage entries in db matching a single file.
* Create Game Information Page
  + GM Page
  + Player Page
* Begin Optimization
  + Finish Switch to use of Master Page
    - Aka fix remaining javascript bugs caused by the switch.
  + Remove JQuery
  + Cleanup CSS usage
  + Disable ViewStates on unneeded pages

Week 6 (Nov 18 – Nov 24)

* Begin Project Testing
  + Get nerd friends to use the application for my D&D games so Players portion can be tested
  + Utilize the Application myself for Game management to help test GM portion
  + Perform specific testing on all Functional Requirements
  + Time page accessing/database operations to see if within limits defined in Non-Functional Requirements
* Finalize any remaining functionality of Main Project
  + Put something in the Website’s Home Page section…
  + Remove Debug Buttons
  + Do something with “Forgot Login?” Label on login screen. Maybe just delete it since no email system exists for a password reset. Too much to learn within time allotted.
* Begin work on secondary project (Desktop app)
  + Adapt any Class/Object changes in DungeonBuddyOnline to C# Desktop App, shouldn’t be very many.

Week 7 (Nov 25 – Dec 1)

* Go around the Website intentionally trying to break stuff, or have Bob do it since nothing ever works when he’s at the controls.
* Complete login/authentication feature of Desktop application
  + Find way to store database credentials securely (encrypt app.config)
* Try to find a way to adapt the Magic Item Shop to work with the online version. May not be doable in time allotted, since the Desktop App can only handle one shop, and the Online can handle infinite shops.
* Complete Desktop application’s ‘Import/Export’ functionality
  + Upload/Download of needed files, hookup to existing serialization methods and .bin files
  + Add UI Screen (Tab) for this functionality

Week 8 (Dec 2 – Dec 8)

* Complete Testing
* Complete Secondary Project

Week 9 (Dec 9 – Dec 11)

* Present Project Demo and Testing on 11th

**Timeline Changes**

The biggest changes made to the timeline were the swapping of the original Week 3 with Week 4, as I realized upon reaching Week 3 that it made more sense to hookup the Create and Join Game functionality first so I could stop using non-dynamic Game objects to test all of my code. I also added Optimization to Week 5 to reflect the Course’s introduction of the material, pushing back the Testing to commence in full Week 6, though truthfully I like to think I’ve spent a good amount of time testing things as I code them.

The pushing back of Testing does raise some concerns, however, in that it means a shared focus between the testing and my secondary project, which starts the same week in Week 6. It isn’t unfathomable therefor that unexpected hurdles in either could arise and reduce time spent on the other.

4/5. Provide a narrative on the optimization features that are in place or ones that you plan on putting into place. Describe why you plan on or why you put those into place.

**Existing Features**

* The obvious optimization feature that has already been implemented is the use of Stored Procedures in the DungeonBuddy database. This allows the SQL commands to be precompiled and thus be faster.
* I also opted to utilize Server Cache for the loading/storing of the Random Name Tables, Random NPC Traits, and list of Magic Items, as they are just a bunch of strings (not very big file size wise) that will need to be accessed in large numbers within a short period of time, making database operations less than desirable. They are also the same values needed by every GM making Cache seem like the fastest and best option.
* Not sure if it counts, but I also utilize a Master page on every one of my pages as of yesterday, and this means they now load WAY better than my crummy object tag implementation, which would load the navbar instantly and the remaining 3/4s of the screen half a second to a second later, with it being a big white box in the meantime.
* Lastly, I recently moved all of my external javascripts to the bottom of the form pages so that they can load last instead of being loaded first. This is so behind the scenes javascript doesn’t hold up the display of html to the user.

**Planned Features**

Additional planned features I expect to implement are:

* Removing JQuery as I think there’s only one single instance of me using it, making its overhead just not worth it.
* I also plan to compact my CSS style sheets down to just two per page (a general style sheet, and a specific), as right now the worst of my pages have four.
* Disable ViewState on several pages that do not need it. I am hoping I can get away with this on all of my Dynamic Table pages, since all of the content has to be pulled from a Session variable anyway since PostBacks destroy dynamic tables. This should reduce load times.
* I will probably minify my css and javascript files eventually once the project is complete and ready to more permanently deploy.

**Not Planned Features**

* I was going to use Server.Transfer, but it broke everything. After fixing what it broke, I discovered that PostBacks still alter the Page URL, which results in a URL that is virtually never accurate to the page being displayed. If I find a way to stop the URL from changing to inaccurate links I may swap it back in, until then it’s not professional enough looking.

6. Describe how you plan on measuring the optimization of your application.

I intend to code in several test methods into the application to time how long certain functions take to complete, for example the dynamic table creation methods, as they seem the most intensive. I also will for sure time the group of methods that load the Random Names and Traits into Server cache, to make sure that does not become a burden on server resources, though I expect it not to be.

Since my website is already running on a Web Server based in New Jersey, I can also test the final product’s ping, and bandwidth usage to make sure the application remains responsive after deployment. Realistically though I intend to start utilizing the application in the field (D&D sessions) tomorrow, so if I find myself switching back to my Desktop version during play then obviously things aren’t good enough, and if I don’t then they probably are.