**The Proposal Document**

The project proposal should have the following sections:

1. Project Name

RPGWebServer

1. Group Size / Contact information

I will be working on the project by myself

1. Abstract/Propose

The goal of this project is to create a database to store and distribute information that is key to running a Table-Top RPG group. This includes the data for Player Characters, Non-Player Characters, Creatures both friendly and hostile, as well as plot relevant information for quick access.

1. Background/Prior Knowledge
   * What do you already know about the topic, technology, or subject you will be working in. Do you consider your self a Newbie, Beginner, Novice

I have taken the SQL database class on campus and frequently needed to access SQL databases for work reasons. I would consider myself a somewhat advanced beginner with these topics.

Additionally, I plan to host this database on a raspberry pi. This will require me to become familiar with the abilities of these devices as well as find a sustainable way to run this device for long periods of time.

* + Provide information essential to understanding your project. If applicable, this should include:
    - Prior work by others - Are you recreating something that has been done before? Are you building on top of others' work?

There are many applications that do similar work that already exist in the world including DnD Beyond, Lion’s Den’s “Fight Club” and Roll20. These applications each have some aspects that I like and some that I do not. I will be referencing their User Interfaces and identifying ways to improve my own project.

* + - Prior work by you - Have you already done things in this area, and you are adding to it?

I have spent some considerable time contemplating the design of this system as it is one that I have been wanting to create for some years. I have some basic design documents as well as some simple mockups of the data. It will take much more work to have a good, fully flushed out database design.

I have also worked with similar user interfaces and applications as the front end of this project in the past and have significant experience as to what works and what doesn’t.

* + Provide information on what foundational course background got you interested in this subject and how you are going to use that information in your project.

The courses that really got me interested in this kind of work were the Database Design and Development, Computer Systems, Computer Architecture and Data Intuition and Insight courses.

* + Provide Initial Research: Spend some time looking through the resources that are already available on your project. (i.e. BYU-Idaho Library, Internet)

**TODO**

* + Do a Topic Web or Mind Map: Create a map of your idea, to other ideas and related fields. This will help you understand your audience, purpose, related application of the idea, and potentially other used of your idea.

**TODO**

1. Description
   * Provide the details of your project. In particular, make sure to include:
     + **Why?** – Reason for buy – Why should the who use it? More important than the what? Facts Tell, Stories Sell. Having a good story as to why you are creating the application is always a seller. It also gives you a starts the profile building of your users. You can’t be the sole user, make an aviator of yourself and use it.
     + **What?**: A Description - In more detail than your abstract, explain what your will project do. What is the solution/features? Does the solution to your problem already exist? If so how is your solution better? Are you re-inventing the wheel? Think of an Info commercial, present your problem, present know solutions and flaws, and convince me why yours is better. What is the real-world impact of your solution? What will prevent them from going back to the old solution or still have the same problems?
     + **Who?**: Target Audience - Describe the intended audience, customer, user of the project. Yes, there is a difference between the customer, who pays for it, and the user, who uses it. What is your primary audience? Who is going to use your solution/program? Are there other markets you could advertise? Will your solution only be in a certain geographic area? community? age range? etc.
     + **Where?** Platform – Where are you going to develop your solution? Where is it going to be used?
     + **How?** Workflow – How will the who use your solution? How will it work?
     + **When?** SMART Goal - When do you know the project is done? When is good enough, or I have a valid product that I can demo. When do I know I am done with one stage of the project and should move on to the next. This is very similar to Achievable for setting SMART Goals
2. Significance
   * Referring back to the expectations for significant projects above, explain how/why your project will be significant.
   * Is this something that you can put on your resume, and you would feel would impress prospective employers? Then describe what you would put on your resume, not that it would be a good resume filler.
   * Note: There is not free version of this solution, this is not a good reason to create your solution, and should not be a factor unless you have an alternate business solution to cover the costs. Many solutions you must pay for, for good reason, they take resources to provide. Yes, software is invisible, and has no substance, but the hardware & energy it runs isn’t. The people supporting the software need to be paid.  There are many different business models that leverage different levels of free. Remember if you got something for free you were not the intended customer.
3. New Computer Science Concepts
   * Another critical part of your senior project is that you demonstrate that you have become a self-reliant learner. Please describe the new things you will need to learn to complete this project. These items should be computer science / software engineering topics.
   * Your sole reason should not be to learn a language. Especially if you have already taken a class in it. It should be a new aspect of the language not covered in of the previous classes found in the major.
   * You should also include a cool new tool, technology, 3rd party software, or programming concept.
4. Interestingness
   * Describe why this project is interesting and exciting to you. Senior projects get hard, hit road-blocks, and cause people to want to quit. If you are excited about your project, this will help you push through.
   * Thinking that the language, tools or technology is interesting is not a good enough reason \*alone\* to justify your project. Too many programming projects have failed, prematurely outdated, or stopped being used because the latest shiny technology was used. You also need to mention why you are interested in the problem domain that your solution falls in. This is where you express your why, different than your audience’s why.
5. Tasks and Schedule
   * Course Expectation is that this is full semester effort of 126 hours/person. That is 9 hours per week. Please lay out your schedule, in 8-10 hour estimations. If you are working in a group use the following equation; effort = people\*126\*90%, where 10% is communication effort involved in working with more than 1 person. For example, 2-person group = 2\*126\*.9 = 226 hours. Show me the total per person and for the group.
   * Divide your project into tasks. Try to make these as meaningful as possible and more detailed than "development." For example, for an OCR project, you might have separate tasks for loading an image into the system, identifying the text region, segmenting into characters, etc.
     + For each task, list the deadline and estimated number of hours for the task. Your schedule should include submitting the requirements specification as a milestone. Be sure to consider holidays and other events that may impact your schedule. Be realistic!
   * Provide the total estimated number of hours to completion.
   * You do not have to specially align with a weekly schedule.
   * A typical project lifecycle is 10% Research & Development (R&D)/Concepts of Operations(CONOPS), 20% Requirements Analysis, 30% Implementation, and 40% Test and Integration. Please plan accordingly. If you are learning something new, make sure you have allocated enough time for the rest of the project. Testing will take a lot longer than you think it will. If you have already done this good job.
   * Is your schedule to aggressive? Do you have more hours than you have time/resources? Are you spending more time for this project than is needed? If you are planning on working more than 20 hours a week per person on this class, you might have to reduce the scope. Also anything over 200 hours/person also might need to be taken look at, which by the way is equivalent to 2-person group. Prove to me that it won’t go over 200 hours/person.
   * How are you going to measure your success and progress? This last year, I starting keeping track of my calorie intake using an app I downloaded. I soon realized with in a day or two what I was eating. Because of the awareness, I started making better choices and higher quality foods, because I was now limiting myself to a budget. From those choices, I started changing my behavior.  How does measurement effect quality? I was once told that just by defining and measuring a process you will see a 70-80% improvement.
   * What evidence will you provide that the task is complete for that week? How do you know that it is done?
   * Please note that this schedule is for your benefit and not something you will be held to for a grade, but rather something we can refer to, to see if we are on schedule or not.
6. Resources
   * List resources needed to complete your project. This may include hardware, software licenses, reference material, etc. Specify the estimated cost for each resource.
     + Include hardware, software, compliers, books, websites, mentors, events, and videos associated with languages, tools, and software you need for the project.
   * Too many projects die before they even get started, because the infrastructure is not in place for the success of the solution. The dependences are not fully explored, and risk analysis is taken. What are your dependencies for success/failure of your project?
   * What are the languages do you need to install? What IDE will you use? What platform (Windows, Mac, Web, Servers) are you going to use? Where are you going to develop and test the solution? How are you going to install & deploy the solution?
   * This is to help you recognize if there are things you'll need to buy, and if so, if that is feasible for you. In some cases the department may have limited funds to purchase equipment that can be reused for future projects. But recognize that if the department purchases equipment it will stay with the department.