**Senior Project Website**

**Version 4**

**Feasibility Study and Project Plan**

CIS 4911 Senior Project

Section U01

Professor Masoud Sadjadi

**Prepared by**

Julio Perez

July 25th, 2014

# 

Copyright

*All title, including but not limited to copyrights, in and to the SENIOR PROJECT WEBSITE are owned by School of Computing and Information Sciences Faculty of Florida International University. All title and intellectual property rights in and to the content which may be accessed through use of the SENIOR PROJECT WEBSITE is the property of the respective content owner and may be protected by applicable copyright or other intellectual property laws and treaties. This EULA grants you no rights to use such content. All rights not expressly granted are reserved by School of Computing and Information Sciences Faculty of Florida International University.*



Abstract

*The purpose of this document is to conduct a feasibility study of the current "Senior Project Website". In this study, the deficiencies and limitations of the system will be identified and solutions will be proposed as part of the new system. These solutions will not only address the opportunities present in the current system, but will also enhance the system by proposing additional components that are significant enough to provide a better user experience to mentors and students. Furthermore, this document contains a description of the team’s general plan.*

**Table of Contents**

*Senior Project Website V4*



[Copyright 2](#_Toc393537026)

[Abstract 3](#_Toc393537027)

[1. Introduction 6](#_Toc393537028)

[1.1. Problem Definition 6](#_Toc393537029)

[1.2 Background 8](#_Toc393537030)

[1.3. Definitions, Acronyms, and Abbreviations 8](#_Toc393537031)

[1.4. Overview of Document 10](#_Toc393537041)

[2. Feasibility Study 11](#_Toc393537043)

[2.1. Description of Current System 11](#_Toc393537044)

[2.2. Purpose of New System 12](#_Toc393537046)

[2.3. High-level Definition of User Requirements 13](#_Toc393537047)

[2.4. Alternative Solutions 15](#_Toc393537048)

[2.4.1. Description of Alternatives 15](#_Toc393537049)

[2.4.2. Selection Criteria 17](#_Toc393537050)

[2.4.3. Analysis of Alternatives 18](#_Toc393537051)

[2.5. Recommendations 19](#_Toc393537052)

[3. Project Plan 21](#_Toc393537053)

[3.1. Project Organization 21](#_Toc393537054)

[3.1.1. Project Personnel Organization 21](#_Toc393537055)

[3.1.1. Hardware and Software Resources 23](#_Toc393537056)

[3.2. Identification of Tasks, Milestones and Deliverables 24](#_Toc393537057)

[4. Appendix 25](#_Toc393537058)

[4.1. Appendix A – Project Schedule 25](#_Toc393537059)

[4.2. Appendix B – Feasibility Matrix 26](#_Toc393537061)

[4.3. Appendix C – Cost Matrix 27](#_Toc393537062)

[4.1. Appendix D – Diary of Meetings 28](#_Toc393537063)

[5. References 29](#_Toc393537064)

1. Introduction

*Senior Project Website V4*



This section introduces the “Senior Project Website: Version 4” pertinent document information. First a problem definition is given regarding the problem statement being addressed in this version. Background on what prior versions up until now will also be addressed. Pertinent particular definitions, acronyms, and abbreviations used in this document will be defined here as well. Finally an overview of the entire document will be included for ease of navigability.

* 1. Problem Definition

For the Senior Project class students are required to join projects that are of interest to them making up a team for a project whose positions are of variable length. The process of creating such a team of course carries many constraints and fulfillment of those constraints while optimizing the speed of generating the team is the primary focus of this version of the Senior Project Website. Thus the primary focus of this version is improving the websites matchmaking capabilities.

**Constraint Fulfillment**

The generation of a team for a project carries many constraints requiring consideration. Those constraints are threefold:

1. The *head professor*, primary manager of the website and course, wants certain projects optimized with regards to having a team that fulfills all of a projects requirements, having students who have the most project-relevant skills for the project, and having students’ skills efficiently distributed whenever possible
2. The *project proposer*, the proposer of a project, can be the head professor, the person who proposed a certain project wants the same thing the head professor wants but only for their project. Can also be seen as “what the project wants”
3. The *student*, the prospective position filler, wants to be assigned a particular project, or if they want, they can declare they do not care which project they are assigned and would like to be in any project. Hence one can say they would like to be assigned a project best fitting their skills in that case.

**Speed Optimization of Matchmaking**

The generation of a team when brute-forced can be seen as trying every combination of a team made from every student for every project. Such a phenomenon has been observed as a “combinatorial explosion” which one can surmise would lead to an incredible large number of computations in search of a “perfect team” for a project fulfilling constraints outlined in the previous part. Mitigation of such phenomena will be a big point of interest in this version of Senior Project Website.

**Head Professor Matchmaking Feedback**

As the coordinator of the course and website the head professor should have final executive decision in regards to what assignment of students to project will end up being. Giving better user interface, better flexibility to the matchmaking process, and some amount of control regarding how matchmaking progresses all would benefit how effectively they can carry out their duties.

* 1. Background

Students enrolled in the Senior Project course work in teams of variable length on projects supervised by the head professor outlined by mentors or the head professor themselves. The student will display knowledge of software development and capability of discussing with a client for the software. In the first week, students select some of those projects.

In previous semesters of this class, students develop the SPW to facilitate this process. In Spring 2013, version 1 was developed with basic functionalities. This team identified the drawbacks of using the Google shared document. They collaborated to design a website to make the signup process easier and equip that process with more integrity. In Fall 2013, another team of students continued the development of version 2 furthering the website’s functionalities. They provided a structure to the process of signing up with a team. They also equipped the system with pleasurable conveniences.  In Spring of 2014, a third team continued development as a version 3 wherein they increased the head professor’s capabilities, implemented an automatic matchmaking process and allowed users the ability to shard documents. In general, the SPW is currently in operation by students, mentors, and the Head Professor of the Senior Project Class.

* 1. Definitions, Acronyms, and Abbreviations

EULA: End-User License Agreement

FIU: Florida International University

Google Docs: Free web-based office suite offered by Google within Google Drive service.

PHP: Open source server-side scripting language designed for web development to produce dynamic web pages.

SPW: Senior Project Website

SPWv.1: Senior Project Website Version 1

SPWv.2: Senior Project Website Version 2

SPWv.3: Senior Project Website Version 3

SPWv.4: Senior Project Website Version 4

**NRMP**: A form of matchmaking devised by the national residency matchmaking program (NRMP). The details of it are that hospitals have residency spots to fill and a ranked list of applicants they want filling them. Applicant ranked residency programs themselves. The matchmaking process works by having applicants apply to residencies in their list’s order. If another applicant tries to displace a current tentative applicant they displace the tentative applicant if the challenging applicant is more desired by the program else they try the next program, displaced applicants try their next program too. Matching ends when all applicants are match, or all applicants are match minus ones who went through their entire list.

**Heuristics**: Colloquially means “common sense approach”, in computer science the meaning is adapted to mean a technique applied to solve a problem. I.e. to shorten average job time in a computer do short jobs first.

**VIP**: Very important project, a project ranked by the head professor to be of a score between 2 and 100 this project will undergo intense matchmaking using heuristics to find a perfect team quickly (higher scored means higher priority)

**OP**: Other project, a project ranked by the head professor to have a score of 1, such projects will undergo NRMP matchmaking to give the ability to compromised project proposer and student constraints. These are more hands-off.

**Agile methodologies:** A method of software development where features are done on an iterative basis with increments on a time period by time period basis. I.E. a developer aims to do feature X this week they do it and they might do more, or they fail to do it and will require more time. Obviously stalling too much on a feature might lead to consequences.

* 1. Overview of Document

Section 1 Introduces the SPWv.4 project goals.

**Section 2** Discusses the feasibility study, meaning how feasible the proposed system is versus already developed technology and other alternatives.

**Section 3** Covers the project plan and organization

**Section 4** is an appendix of suplimentary information for prior sections.

**Section 5** covers the references in this document.

1. Feasibility Study

*Senior Project Website V4*



The feasibility study analyzes alternative solutions to the problems. For each alternative solution, we will examine its benefits and limitations. With this study, we will be able to determine the best possible solution after a detailed analysis. We will describe the current system and the purpose of the proposed solution. The study will conclude with justifiable recommendations.

* 1. Description of Current System

The current system requires a big update with regards to the matchmaking process of students to projects. The prior implementation is very slow falling into the trap of combinatorial explosion described in section 1.1 “Speed Optimization of Matchmaking”. It is slow to the point of never returning given a “fair” size of data, as in 8 projects and 24 students, instead just timing out.

The system also tries to fulfill the constraints of the matchmaking system by producing 10 different matches which considers the constraints outlined in section 1.1 as one score resulting in something not completely discrete in regards to function. More importantly matchmaking cannot run in an imperfect environment such a condition is rather unreasonable as it is not prudent for the head professor to discount a project because no students have one skill one project needs. A perfect matching after all is not always possible, nor reasonable to coerce.

The result page of the matchmaking process is also lacking in usefulness. The display lacks useful information and it automatically updates the database without the head professor doing anything other than ranking projects by their opinion of importance. Such disconnect between the head professor and his executive duty in matching students to projects makes the usefulness of the current system lacking.

* 1. Purpose of New System

The new system in SPWv.4 aims to build upon various infrastructures afforded to SPWv.4 by SPWv.3 (like database changes, head professor rankings). Furthermore SPWv.4 aims to completely revise the core of the matchmaking process.

The three constraints will be addressed by partition the needs in various ways. For example the head professor’s desires will be paramount as he will declare very important projects (VIPs) which will undergo matching in the order outlined by their rankings and undergo intense matching to maximize all the projects skill needs while fulfilling them all (i.e. have project’s Java, C, C++ needs met first then try to get all students to have all of those skills). These VIPs of course benefit their project proposers. In VIP the head professor can choose between students who want those projects versus all students, thus giving him ability to balance student’s interest versus their own. The matching will be two-phased and the second phase will occur on other projects (OP), projects the head professor does not particularly worries about, which will work by allowing students to choose the project they want by their interest but displaced when a student wants it more or a student better fits the project, again the head professor has an option of customizability and balance here.

Speed will be increased as VIP will work by heuristics to mitigate some of the headache of combinatorial explosion by searching “intelligently”. Heuristics will cut down searches by not looking into impossibilities like the first position in a project with 10 skills will never be filled by someone with no skills (unless they are the only person), and also by not search logically skill equivalent people. OP will work by the national residency matchmaking process which is not a combinatorial team search rather goes through every student’s list thus as worst takes the amount of time to see every student multiplied by the time to see their entire list (which at most is the total amount of projects).

Customization will increase by what was stated early shallow choices for OP matching and the possibility of harder choices for VIP if ran project by project. The display will be improved to show more feedback regarding how students affect an overall match result, a project match result, and their contribution to a project.

Overall this system should allow the head professor to decide how to balance the three constraints of the system, be faster on average, and allow more customizations with regards to what a match does in addition to more data feedback.

* 1. High-level Definition of User Requirements

**Version 1:**

* Project members can edit their project.
* Project members will be able to invite students to their team.
* Project members will be able to invite other users as mentors to their team.
* Project proposals will include the technologies to use, in a tags-like way.
* The Head Professor will be able to approve or reject projects.
* Users will be able to register using their LinkedIn accounts.
* User will be able to pull their technology skills from LinkedIn
* The system will automatically present users with a list of suggested projects
* The system will be able to present users a list of suggested users (as mentors or students) for their projects
* Users will be able to search for users, skills and projects.

**Version 2:**

* Head Professor will be able to set the internal system deadline.
* An API will be implemented to allow access to the site only to students registered on Senior Project Class , create/update/remove d students profiles and provide info about students and their projects titles to the Mobile Judge and Online Job Fair projects.
* Guest access will be provided for those students who are not registered on the class.
* Registered users will be able to upload profile pictures from local storage.
* Student will able to upload, download files to the set of shared project files.
* Mentors will be able to upload, delete or download any file that has been uploaded to the server.

**Version 3:**

* Best Fit Project Algorithm: calculates the best possible student-project match.
* Project Repository, where the Head Professor has full access to all projects.
* Users may share documents with project scope.
* Users may download a zip folder of the entire project.
* Head Professor may use the Files Repository without the use of milestones.
* Manage Users: create new mentors manually, or send an invitation email with a unique link for verification. Edit or delete existing users in any role, especially inactive students and mentors from previous semesters.
* Head Professor can override users on SPW.
* Allow the Head Professor to impersonate any user and act on his behalf.
* Aesthetics and Efficiency Fixes:
* Add side scroll arrows to view full description of all projects sequentially.
* Modify login window format.
* Give SPW a responsive design for optimal viewing across a wide range of devices from mobile phones to computer monitors.

**Version 4:**

* Partition the defined constraints of matchmaking
* Increase matchmaking speed
* Give better feedback to matchmaking’s effect to head professor
* Give some more customization options in matchmaking for the head professor

* 1. Alternative Solutions

This section will describe alternatives solutions to the problem. A selection criterion will be analyzed based on the feasibility of such alternatives and finally all the alternatives will be compared concluding on the best option to pursue.

* + 1. Description of Alternatives

Alternative 1: Current Usable System

The current usable system does match students to projects and has set up great infrastructure to support it. But the core of the matchmaking is very slow around the O(2^n) or O(n!) which is highly impractical for a decent amount of data input. Furthermore the matchmaking occurs without any confirmation once ran and can use more feedback with regards to what it has done. Thus to an extent the current system is rather lacking in real application.

Alternative 2: Start from Scratch

Starting from scratch on the matchmaking system is rather wasteful as the infrastructure for allowing students and the head professor to rank projects is rather solid. The only benefit to this would be if the projects were started from scratch to allow definition of projects to require “3 Java skills, 2 C skill…” teams to better define team needs to an exact. However such a thing might not be specifically useful as most projects would likely want to have all students in their team to have all the project’s skill needs. But this might be of consideration in a future time.

Alternative 3: Proposed System

The proposed system’s strength is in the discrete nature of dividing student’s interest, the head professor’s preferred projects, and the project proposers desire for optimal project teams. The proposed system also applies heuristics to “intelligently” find perfect teams for projects. The drawback is that the intense matchmaking aims to improve the projects optimally one by one which may lead to the global match of VIPs to be less impressive if for example two students considered for project 1 if the slightly worse one is chosen the slightly better would greatly improve a later project. How often that would occur would require more research into probabilities then that can be afforded.

Alternative 4: Research and Application of a More Composite Approach

Alternative to the discreteness of the proposed system is the application of a more composite one. A possible way to do matches in a composite environment is to do A\* search however A\* requires an idea of “direction” and in a composite environment where project skill maximization/fulfillment, student’s interest, and head professor’s interest are abound such a direction would be hard to pinpoint. The benefit of researching this more and solidifying a manner to apply this is that overall match data would be maximize of course these elements work against each other and would require balance and leeway to one metric or another. Such application and research however likely will take more time and expertise then the Senior Project class can provide.

* + 1. Selection Criteria

The previous alternatives will be analyzed based in terms of their operational, technical, schedule, and economic feasibility. Following criteria utilized from SPWv.3 Feasibility study.

Operational Feasibility

The operational feasibility deals with how functional, how well an alternative deals with the system requirements and how well it is received. An ideal solution would be user-friendly and well received. Is it easy to use? Does it solve the problem? Does it provide the users with accurate and desired information? Does it generate positive feedback by its users?

Technical Feasibility

The technical feasibility focuses on understanding the available technical resources and how they can be used to implement the system. Is the builder team familiar with the technologies? How well do they need to master the technologies? Are the technologies supported at the moment or/and in the future?

Schedule Feasibility

The schedule feasibility deals with the amount of time needed to implement an alternative. An ideal alternative will be quick to produce. How long will this solution take to design and implement?

Economic Feasibility

The economic feasibility analyzes cost and benefits of an alternative. An ideal alternative will be inexpensive to produce. How much will it cost to implement? Will there be any gains? Will it have additional costs in the long run?

* + 1. Analysis of Alternatives

In regard to the metrics outlined in 2.4.2 analysis of the alternatives under those metrics are as followed.

Alternative 1 the current system is rather inoperable taking too long to run matches, and lacking user interaction/useful data. Alternative 1 of course being built would require no further technical nor schedule concerns and would cost nothing to build, however economically speaking the utility of the current system is very lacking despite its useful infrastructures.

Alternative 2 starting from scratch may lead to a slightly better reception then alternative 3, the proposed system which works using utilities from the current system. Starting from scratch can provide some more customizability to skills a project requires though but to what extent a project proposer really needs to outline “how many” of a certain skill they need is a questionable need, particularly if most answer “as many as possible”. If project proposers do provide more accuracy as a result of accurate pinpoint skill needs better match results may be possible from alternative 3. Technically speaking research into database alteration would be required and lots of time to change the current infrastructure leading economically to less likely chance of getting all of what’s needed finished.

Alternative 3, the proposed system, builds upon the current system thus bringing utility from prior efforts spent. Alternative 3 plan to increase utility to the head professor’s role in matchmaking hence increasing operational feasibility. Technically and schedule wise the technologies will require lots of research in order to use but regarding backend its technically feasible; schedule wise time will be require to tweak user interface to something useful. In regards to economics the tool will be of better utility albeit alternative 4 might be considered better.

Alternative 4, a composite approach, whiles potentially be great in operational feasibility requires an extreme amount of technical and schedule requirements to figure out how to technically implement the alternative and requires a lot of time to do that research at this level. Economically this lead to very little actual product if the research ends up fall apart or not working to a specific specification.

* 1. Recommendations

The SPWv.4 version therefore recommends alternative 3, the proposed system. The proposed system is of utility as it uses infrastructure from SPWv.3 and is feasible for a student to complete in the span of this project. This system will provide utility as well. Alternative 4 while it might have more payoffs the cost in designing and implementation might be too high to be feasible at all for now and may in fact result in a mixture of lacking utility with regard to focus, as it may not be possible to balance the constraints within one metric.

1. Project Plan

*Senior Project Website V4*



This chapter will cover the project plan for the SPWv.4, personnel organization, hardware and software requirements, milestones of the project and the cost involved on its development.

* 1. Project Organization

This section details how the SPWv.4 will be organized throughout the life of the project. It also contains the technical needs of the project at this time. This includes the hardware and software needs of the developers.

* + 1. Project Personnel Organization

This is a one-person project spanning 12 weeks, all done under agile methodologies, where incremental weekly goals and updates are made and attempted to be met. Meetings therefore occur weekly to with the client, team meetings are of course of no utility to a one person team, daily weekday meetings with a supervisor/mentors however are done for progress updates and advices requisition. Table 3.1 shows the roles of the team and table 3.2 shows the roles in project development/preparation.

|  |  |
| --- | --- |
| Name | Role |
| Julio Perez | *Team Leader:* Responsible for ensuring the project progresses based on specifications from client and advice from mentor. Leads discussion ideas and adapts to client needs. |

**Table 3.1** – Roles

|  |  |
| --- | --- |
|  | Julio Perez |
| Project Plan | ✓ |
| Deliverables | ✓ |
| User Interface/Views | ✓ |
| Database |  |
| Controllers/Models | ✓ |
| User Authentication |  |
| Testing | ✓ |
| Final Presentation | ✓ |
| Demo | ✓ |

**Table 3.2** – Project preparation and development roles (Note: “✓” denotes participation in role)

* + 1. Hardware and Software Resources

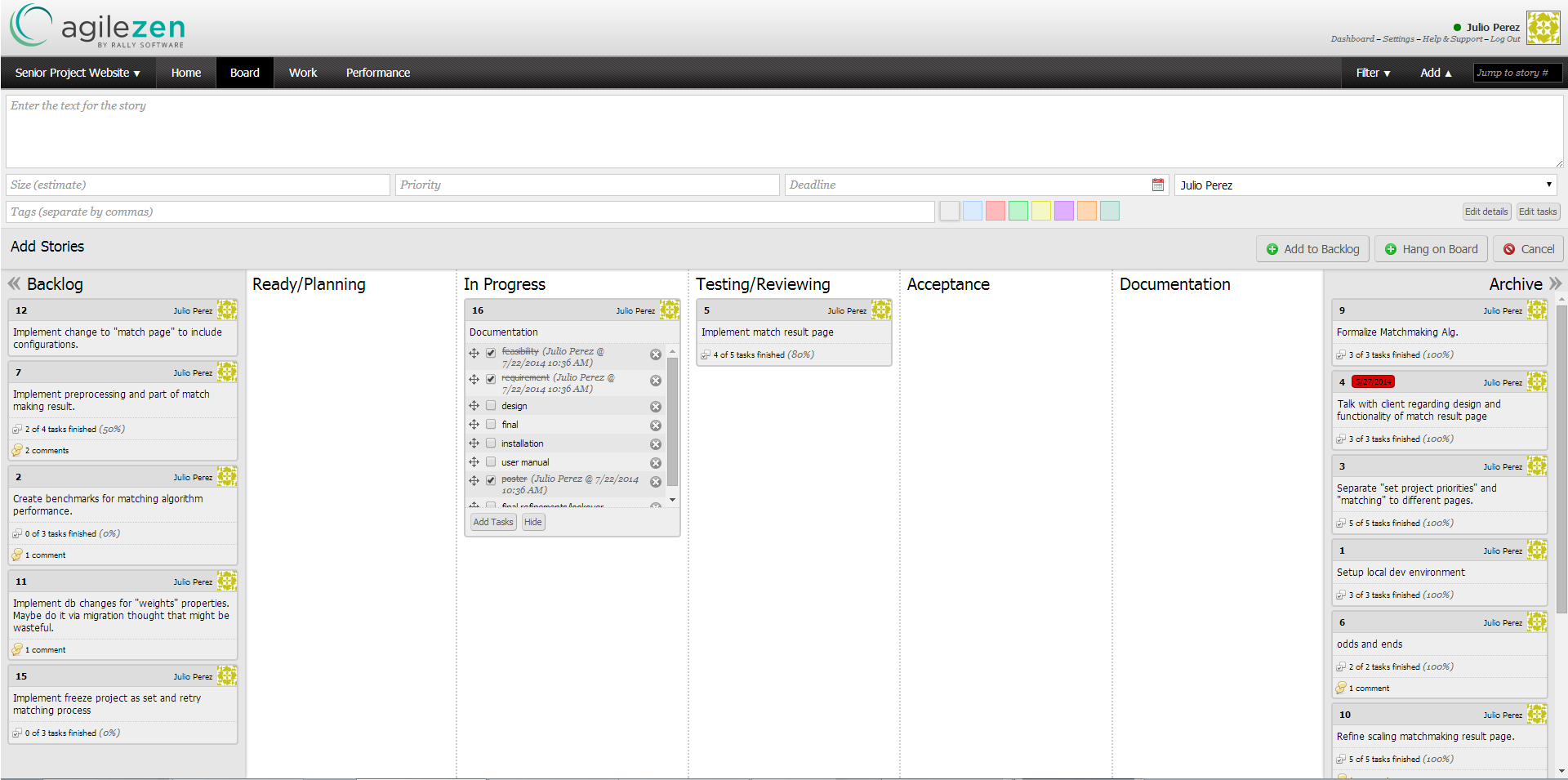
This section lists the software and hardware needed to develop and deploy the proposed project. Hardware and software will not change from version 1 to version 4.

|  |  |
| --- | --- |
| Software | Hardware |
| Sublime Text 2 | Processor: x86 / x64  2.0 GHz or faster |
| PHP 5.1 | Memory:  2 GB DDR3 or higher |
| MySQL 5.0 | Disk Space:  20 GB |
| Microsoft  Office Package 2010 | Display:  1024 X 768 with DirectX 9 |
| Mac OS X |  |
| Microsoft Windows 7  Home/Pro/Ultimate |  |
| Apache 2 |  |
| Firefox Mozilla  ver. 16.0 or higher |  |
| Internet Explorer  ver. 7.0 or higher |  |
| Safari  ver. 5.0 or higher |  |
| Google Chrome  ver. 19.0 or higher |  |
| Skype / Team Viewer |  |
| GitHub repository |  |

Note: This table is from the SPWv.1 “Feasibility Study & Project Plan” (Fernandez, Sanchez, Moya 1).

* 1. Identification of Tasks, Milestones and Deliverables

Task and milestones have essentially for this semester occurred on a week by week basis. The end result being the matchmaking system outlined in this document proposal. Task essentially boiled down to week by week tasks and debugging with a general milestone of achieving the overarching goal as well as defining the goal as weeks go on. Deliverables are to all be turned in at the end. Deliverable relevant content also is to be gained during the weeks.

Project management meetings, tasks, and such generally define by Agile Zen a tool to aid in this agile methodologies approach new to this semester. Pictured below.

1. Appendix

*Senior Project Website V4*



* 1. Appendix A – Project Schedule

Taken from the given syllabus and adapted to what actually occurred.

### Schedule

|  |  |  |
| --- | --- | --- |
| Week | Date | Topics |
| 1 | 5/13/14 | Project Assignment and Grouping |
| 2 | 5/20/14 | Design & Goal Refinement |
| 3 | 5/27/14 | Design & Goal Refinement & Debugging & Code Refinement |
| 4 | 6/3/14 | Design & Goal Refinement |
| 5 | 6/10/14 | Design & Goal Refinement |
| 6 | 6/17/14 | Design & Goal Refinement |
| 7 | 6/24/14 | Design & Goal Refinement |
| 8 | 7/1/14 | Design & Goal Refinement |
| 9 | 7/8/14 | Debugging & Code Refinement |
| 10 | 7/15/14 | Debugging & Code Refinement |
| 11 | 7/22/14 | Debugging & Code Refinement & Deliverables |
| 11 | 7/25/14 | Final Presentation & [Demo](https://moodle.cis.fiu.edu/v2.1/mod/assignment/view.php?id=20135) & Poster & Approval Test |

* 1. Appendix B – Feasibility Matrix

Discussion of the following matrix was described in section 2.4.3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Alternative 1: Current Usable System | Alternative 2:  Start from Scratch | Alternative 3:  Proposed System | Alternative 4:  Composite Approach |
| Economic Feasibility | 0 | 9 | 8 | 10 |
| Technical Feasibility | 10 | 5 | 8 | 0 |
| Operational Feasibility | 2 | 9 | 8 | 10 |
| Schedule Feasibility | 10 | 7 | 9 | 0 |
| Total | 22 | 30 | 33 | 20 |

Alternative 1 lacks utility though it is the easiest to implement as its already been implemented. Alternative 2 can provide some more utility but requires more time and schedule concern for rather little benefit. Alternative 3 has high metrics overall but its utility is not perfect, but it is the most feasible overall. Alternative 4 while could be the best would require more technical expertise and schedule than even possible.

* 1. Appendix C – Cost Matrix

|  |  |  |
| --- | --- | --- |
| Project Component | Description – Cost would not be an issue because students are improving on an existing solution to the Senior Project course student, professor, and mentor interaction | % of Total Project Cost |
| Project Management | 0 (Students are working all aspects of the project) | 0 |
| Hardware | 0 (already in place) | 0 |
| Software | 0 (open source) | 0 |
| Testing | 0 (testing tools open source/temporary free license products) | 0 |
| Training | 0 (Students are learning on their own) | 0 |
| Risk Management | 0 (No risk management budget is allocated) | 0 |
| Total | 0 | 0 |

* 1. Appendix D – Diary of Meetings

Given that the team consists of one person no official meeting occurred. Meeting with client and lead developer occurred weekly resulting in the proposed implementation generally described in this document.

1. References

*Senior Project Website V4*



Christopher Kerrutt, William Marquez, Cynthia Tope. "*Feasibility Study & Project Plan for Senior Project Website: Version 3*" (Spring 2014). Florida International University.

Fernandez, Yaneli, Camilo Sanchez, and Keiser Moya. "*Feasibility Study & Project Plan for Senior Project Website: Version 1.*" (Spring 2013). Florida International University.

National Resident Matching Program “The Match” Retrieved January 25th 2014, from http://www.nrmp.org/match-process/match-algorithm/

COMBINATORIAL EXPLOSION. (n.d.). Retrieved July 19, 2014, from Web Dictionary of Cybernetics and Systems: http://pespmc1.vub.ac.be/ASC/Combin\_explo.html

Arbelaez, A., Hamadi, Y., & Sebag, M. (n.d.). Online Heuristic Selection in Constraint Programming. Retrieved July 07, 2014, from research microsoft: http://research.microsoft.com/en-us/people/youssefh/search-socs.pdf