The objective of this project is to develop a simple software product that emulates an actual learning management system, like Blackboard or Moodle.

BlackBoard

12/02/2018

Soumya Srivastava, Kellen Hill, Kevin Nguyen, Abagail Moxon

Table of Contents:

1. Project Background and Description………………………………………………………………… 2
2. Project Programming Details……………………………………………………………………………..2
3. Assumptions and Constraints…………………………………………………………………………….2
4. High- Level Requirements………………………………………………………………………………….3
5. Analysis…………………………………………………………………………………………………………….3
6. Deliverables………………………………………………………………………………………………….......3
7. Visuals…………………………………………………………………………………………………………...4-6
8. Schedule Summary…………………………………………………………………………………………….6
9. Project Organization…………………………………………………………………………………….....6-7
10. Roles and Responsibility…………………………………………………………………………………….7
11. Implementation Plan…………………………………………………………………………………........7-8
12. High-Level Timeline/Schedule……………………………………………………………………………8
13. Resource Allocation……………………………………………………………………………………………9
14. Control Plan……………………………………………………………………………………………………….9
15. Risk Management Plan……………………………………………………………………………………….9
16. Technical Process Plan……………………………………………………………………………………...10
17. Supporting Process Plan……………………………………………………………………………………10
18. Additional Components…………………………………………………………………………………….10

CS 3321 Software Project: Blackboard

Date

# Overview

## Project Background and Description

|  |  |
| --- | --- |
|  | The objective of this project is to develop a simple software product that emulates an actual learning management system, like Blackboard or Moodle. This program will have two modes: 1) the user mode which stores and retrieves student’s partial information in only the current semester and other basic information including student’s name, student’s ID, registered courses (in only the current semester), exam scores in each course, GPA calculation in the current semester and 2) the administrator mode which gives access to insert, update, and monitor the whole system. |

## Project Programming Details

|  |  |
| --- | --- |
|  | Programming Language: Python 3  IDE: PyCharm  Data Storage Method: Python binary blobs with pickle  GUI build with TKInter  Development Lifecycle: Rapid Prototyping |

## Assumptions and Constraints

|  |  |
| --- | --- |
|  | * The deadline must be met. * The product must be reliable. * The architecture must be open so that additional functionality may be added later. * The product must be user-friendly and understandable. * The product must have two modes (administrator and user). |

## High-Level Requirements

|  |  |
| --- | --- |
|  | The new system must include the following:   * Store and compare student and teacher login information * Retrieve student courses and their own grade. * Students can view their attendance record. * Calculate/View GPA * Teachers/Admin can add new grades for the particular student * Teachers/Admin can add new student to the system. * Storage/Retrieval for students |

## Analysis

|  |  |
| --- | --- |
|  | We needed to build a program that initialized a series of students and teachers and save them in a data storage. The students and teacher’s objects will contain their first, last name, password, and ID. Students will also need to be able to access their grades and courses and no modification can be made. To do this we would need a function that creates a variety of student objects and save them to an array. For teachers/admin we need a function that lets the teacher make changes to student’s grades, attendance and add/drop students. |

## Deliverables

|  |  |
| --- | --- |
|  | The complete product will be delivered on December 2nd, 2018. A presentation will be included, on September 29th, 2018, which explains the details of how to manage the product and the overall summary of the project. The project will be delivered in a zip folder to the client. |

## Visuals

|  |  |
| --- | --- |
|  | This flow chart below displays all the things students are able to do.    This flow chart below displays all the things teachers are able to do.    A screenshot of a cell phone  Description generated with high confidenceUse Case Diagram  A screenshot of a cell phone  Description generated with very high confidenceCRC Cards  Flowchart for login & overview  A close up of a logo  Description generated with very high confidenceA close up of text on a black background  Description generated with very high confidence |

## Schedule Summary

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  | | --- | --- | | Type of Flow Chart | Members Involved | | Requirements Workflow | All members | | Analysis Workflow | All members | | Design Workflow | All members | | Implementation Workflow | All members | | Testing Workflow | All members | |

## Project Organization

|  |  |
| --- | --- |
|  | **External Interfaces**. All the work on this project will be performed by four members. The team will meet every Tuesday to discuss progress on the product.  A screenshot of a cell phone  Description generated with very high confidence**Internal Structure**. The choice of our team model is the chief programmer model. In this model Kellen was given the role of chief programmer, since he had the most experience in python language. Abigail was given the programmer secretary where she tested the programs and debug them. Kevin, and Soumya was given the roles of programmers. |

## Roles and Responsibilities

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  | | --- | --- | | Responsibilities | Name of the person In charge | | Documentation | Soumya | | Login/Permission | Kevin & Kellen | | Storage/Retrieval | Kellen | | GUI | Kevin/Kellen | | Data ops | Kellen | | Testing | Abigail & Kellen | | Presentation | Everyone | |

## Implementation Plan

|  |  |
| --- | --- |
|  | **Estimation Plan.** The total development time is estimated to be around 12 weeks and handed over to the client on December 2nd 2018.  **Staffing Plan.** We all agreed to meet weekly to discuss the progress of our parts. Kellen will be needed as project leader throughout the 10 weeks to monitor progress and keep the team on track. The rest of us will collaborate for the first 7 weeks on the initial creation of the product while Kellen and Abigail will work together on the last 3 weeks on implementing and testing the product as programmers.  **Resource Acquisition Plan.** All necessary hardware and software for the project are already available. The application will be programmed in Python.  **Project Staff Training Plan.** After Kellen provided us an overview of how python works, no additional staff training is needed for this project as it will be user-friendly and understandable. |

## High-Level Timeline/Schedule

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  | | --- | --- | | Week 1 | (Completed) Reviewed the requirements set by the client and assigned roles and responsibilities; determined the language of the product. | | Week 2 | (Completed) Produced and inspected analysis artifacts; created software project management plan; shared artifacts with development team | | Week 3 | Produced and inspected design artifacts | | Week 4 | Finish up our individuals codes. | | Week 5 | Finish up our individuals codes. | | Week 6 | Finish up our individuals codes. | | Week 7 | Finish up our individuals codes. | | Week 8 | Finish up our individuals codes. | | Week 9 | Finish up our individuals codes. | | Week 10 | Combine all codes and makes sure there are no errors. | | Week 11 | Testing the codes. | | Week 12 | Finish up the paperwork. | |

## Resource Allocation

|  |  |
| --- | --- |
|  | Kellen will guide the team for the first five weeks, so if there are any problems regarding using python, it can be resolved early. He’ll also check in periodically to make sure the team is on track. Then the team will spend the next six weeks to develop the codes. Then team members will communicate over group me and GitHub to set up the product’s foundation. The remaining three weeks will have focused on making sure the codes have no glitches and all work smoothly together. Maximizing user-friendliness will be of Kellen top priorities as well as meeting the client’s major requirements of GPA calculation. Abagail will be responsible for creating test cases. Documentation and coordination will be done by Soumya. Kevin will work on the login screen and the GUI. |

## Control Plan

|  |  |
| --- | --- |
|  | Any major changes that affect the milestones must be approved by Kellen and documented by Soumya. No outside quality assurance personnel are involved because the rest of the team will be available for testing and checking each programmer’s work. Soumya and Kellen will be responsible for ensuring that the project is completed on time as well as well-documented. This will be accomplished through weekly meetings with the necessary team members (usually all team members are present). Each meeting consists on each workflow’s progress and issues. Kellen will then determine if the team is progressing and following the specification document and the project management plan. Any major problems faced by team members will be immediately reported to their superior who is Kellen. |

## Risk Management Plan

|  |  |
| --- | --- |
|  | The risk factors and the tracking mechanisms are as follows:  - The application should be designed neatly so a student can access the appropriate function even though they will be experienced with computers.  - Because of the ever-present possibility of a major design fault, extensive testing will be performed during the design workflow. Also, each member will test their own code to make sure everything is working properly. Abagail will be responsible for integration testing and, with Kellen, final product testing. |

## Technical Process Plans.

|  |  |
| --- | --- |
|  | Process Model. The iterative-and-incremental life cycle model will be utilized.  Methods, Tools, and Techniques. The workflows will follow the iterative-and-incremental life cycle model and implemented in Python.  Infrastructure Plan. The product will be developed using Python  Product Acceptance Plan. Acceptance of the product by our client will be achieved by following the steps of the selected process model. |

## Supporting Process Plan

|  |  |
| --- | --- |
|  | Configuration Management Plan. Python and GitHub will be used throughout the artifacts.  Testing Plan. The testing workflow will be performed mainly by Abagail with test cases coming from Kellen.  Documentation Plan. Documentation will be edited as the team cycles through iterations of changes. This all will be recorded by Soumya.  Quality Assurance Plan and Reviews. Each member will test the code as a whole with all test cases while Abagail and Kellen conducts integration testing.  Problem Resolution Plan. If any issues arise, members are to report to Kellen. |

## Additional Components.

|  |  |
| --- | --- |
|  | There is a password hide password.  One of our group member had no contribution towards the project because they either dropped out or did not want to do the work. The group had tried to reach the member multiple time, yet there no response. Together the group decided to take the member name off the project. |
|  |  |