Visual Course Planner

Project Charter

Version 0.2
10.14.2018

Table Of Contents

1 INTRODUCTION

1.1 PROJECT INFORMATION

Project: Visual Course Planner for Computer Science
Project Sponsor: Dr. Abdallah Mohammed, Department of Computer Science, University of British Columbia, Okanagan
Project Manager: Noman Mohammed

1.2 PURPOSE OF PROJECT CHARTER

The Visual Course Planner project charter documents and tracks all necessary information required by the client and project team to approve the project. The project charters includes the needs, scope, justification and resource commitments to proceed or not proceed with the project. This document was created during the initialization of the phase of the project.

The intended audience of the Visual Course Planner project charter is the project client, project sponsor, and project team.

2 PROJECT AND PRODUCT OVERVIEW

The Visual Course Planner is a web application that allows students to visually organize their courses for the upcoming year(s) for a specific degree or program offered at the University of British Columbia Okanagan.

The creation of a Visual Course Planner will take place over the next 8 months, starting September 18th and finishing April 1, 2019. Upon completion, students will be able to create an account and use the program to plan out their course schedule for their complete duration at UBC Okanagan. To create an account each user will be required to supply their name (first and last), email, and a password. Password and email recovery is not implemented. First view of the planner will provide students with a default schedule based on the courses offered that year, tailored to whichever program they choose. The ability to move and change courses will be available for each student in a drag-and-drop fashion. Suitable warnings will be visible when a change to the default schedule is made. For example, when the change prolongs degree completion, prerequisites are needed, or if courses clashes with each other (ie. course occurrence at the same time). The program may also perform actions to fix a problem with the course plan (e.g. when a student moves a course, the program may move its prereqs so that they are allocated to an earlier semester). Students will be able to save up to 10 alternative plans on their account so they can leave and return to revise the the degree plan later. Adding Co-op, GoGlobal and summer courses to their degree plan will be an option, Their degree plan will update accordingly - whether that prolongs or shortens the degree completion, add or removes prerequisites. In addition, students will also be able to favourite (via a click of a button) and name their already saved plans for easy future reference. Adding a short memo to each saved plan will also be available.

Once students have decided which courses they want to take during their time at UBC Okanagan, students will be able to utilize a one-click optimization feature which will compress their degree length to the shortest period possible based on a set of preferences and constraints (e.g., taking summer courses, enrolled in the co-op or Go Global program, or limited number of courses etc.).

Administrators of the visual course planner will be able to upload files to inform the system of currently offered courses, degree requirements, and courses that will be offered in the future.

Course requirements and default plans will be pulled from the UBC Okanagan academic calendar.

No budget is applicable to this project.

3 JUSTIFICATION

3.1 BUSINESS NEED

Students at UBC, especially those new to the university, often have a difficult time seeing how courses fit into their overall degree requirements. At present, there is only a static web page that shows which courses are required for a given program without providing the course interdependencies. That information must be found elsewhere. Due to this, the information partition results in confusion around what courses to take during what semester to fulfill degree requirements.

4 SCOPE

The aim of this project is to create a desktop application that visualizes the course structure of a given academic program (e.g.. Computer Science majors in this case). Courses will be interlinked to show course interdependencies (e.g. prerequisites and corequisites). All courses will be distributed over several semesters (default 1 -4 years) that cover the program duration. Once a user (student) creates an account, the system will provide an initial course plan (as recommended by the department) depending on the chosen degree. Users will then be able to modify the plan to better suit their needs. Once done, the user can save their plan under their account so they can retrieve and revise it later if needed. The purpose of this project is to help students better plan out their course schedule for each year at UBC Okanagan.

- are you responsible for Testing impact? - Live deployment? - maintenance? } clarify here

4.1 OBJECTIVES

The objectives of the Visual Course Planner are as follows:

1. Improve students' experience - especially new students - by streamlining their degree planning process and giving them a clear course progression through their degree.
2. Help academic advisors give proper advice based on a students degree plan

4.2 HIGH-LEVEL REQUIREMENTS

a ~~test~~ head in would be nice so the Reader knows whats coming up..

The following requirements that are presented are the projects product, service, or result must meet in order for the project objectives to be satisfied.

- Users are able to create an account and store up to 10 degree plans OR
- Account holders can view, modify and optimize their saved degree plans OR
- A department recommended course plan is initially displayed to the user ← from where?
- Administrators can upload files into the system with information about degree requirements, program structure, and future courses offerings
- If administrators update program/course description, system will automatically check all saved plans and show warning to users as necessary.
- Users are warned of any course clashes, prerequisite conflicts, prolonged degree completion, or missing credits OR → flip → for FR. Perspective
- System will store course information including course description, credits, and course code based on the academic calendar → make sure to address in NFR.

this is a UR

look at this it from SW perspective and from user perspective

re the system will maintain a department Recommended course plan...

4.3 NON-FUNCTIONAL REQUIREMENTS

The following non-functional requirements specify how the Visual Course Planner will work upon completion.

1. Changes must be saved by student before leaving their active session in order be returned to and make more changes.
2. Program information is required before any courses are displayed.
3. Degree information is able to be uploaded.
4. Shortest degree plan is suggested based off of which courses are offered the current year and which courses the student has tailored to their preferences.
5. Warnings are to be displayed immediately once a change has been made that affects the degree plan by prolonging the graduation date, missing prerequisites and course clashes.
6. Drag and drop organization of courses that immediately updates the degree plan.
7. Course information will be stored in a database.
8. The Visual Course Planner will be general purpose and can be used by students of any faculty
9. System will be efficient with processing requests (minimal loading times)
10. System will be scalable and maintainable to accommodate potential growing user base.
11. System will not break down in case of exceptions and errors and assist users to resolve it by using non-technical jargon
12. Users can use this tool via different operating systems
13. Users can use this tool with different browsers
14. The system would be able to deal with surges of traffic without affecting the performance
15. The system will display a warning summary including credits required
16. User data will be protected according to UBC privacy and protection standards[1]

4.4 FUNCTIONAL REQUIREMENTS

The following functional requirements are specify what the Visual Course Planner will do upon completion.

1. Each account stores up to 10 degree plans.
2. Each student is prompted to select a program upon creation of new degree plan.
3. A default degree plan is displayed in the Visual Course Planner based on the chosen degree .
4. An optimize button will condense the user chosen degree as much a possible via amount of years.
5. A warning will be presented if a change to the degree plan prolongs the duration of degree, or prerequisites are needed.
6. Drag and drop functionality for the reorganization of courses based on the students preferences.
7. The optimization will be able to account for student preferences as parameters.
8. Users can save a text description with every course plan.
9. Users can favourite their course plans
10. A menu that appears when user clicks on a course module in the visualization tool. This menu displays additional course information such as course description, credits & a link to the course page on SSC
11. A smart and robust system that adjusts the course plan if encountered with any errors/limitations
12. Arrows of different colours and structure to identify a course's characteristics (prereq, coreq, etc)

[1] https://cio.ubc.ca/sites/cio.ubc.ca/files/documents/standards/Std%2013%20Securing%20User%20Accounts.pdf

4.5 USER REQUIREMENTS

1. The system will accept a user's email, name, and password for account creation
2. Users will be able to save up to 10 degree plans
3. Users will be able to drag and drop courses to make their degree plan
4. There will be an "Optimize" button that a user can click to compress their course load into the shortest time allowed by the school.
5. Administrators can upload current course and program information
6. Users can favourite course plans, displaying them above their other course plans
7. Users can add a text description to each of their saved course plans

4.6 TECHNICAL REQUIREMENTS

1. The system must be able to be displayed on a desktop web browser

4.7 MAJOR DELIVERABLES

MAJOR DELIVERABLE	DELIVERABLE DESCRIPTION
Account Creation	Users are able to create, modify and save their potential degree plans
Course Recommendation Upload	The department will be able to upload a recommended degree plan file. Students are able to build their customized degree from the department recommendation.
Optimization	With one click, students will be able to compress their degree to the shortest amount time, given options like if they are taking summer courses, or are participating in the co-op program.
Course Display	Users can see all currently offered courses for the year the user is creating their degree plan.
Drag and Drop	Users can easily drag and drop courses to build their degree and see any course dependencies (prerequisites, corequisites, standing, minimum grade)
Interactive GUI	Have a visually pleasing and intuitive interface. Easily to understand regardless if the user is a first time user or one that has used to program before; understandable by students by all programs offered at UBC Okanagan campus.

4.8 BOUNDARIES

1. The connection of each students UBC account and visual course planner will not be implemented.
2. Integration with another system (ie. UBC Okanagan website) can be supported using our API but will not implemented during the creation of the Visual Course Planner.
3. Responsive design will not be supported; only a desktop version of the Visual Course Planner will be implemented.

4. The ability to print or export each degree plan created by a student will not be provided.
5. Personalizing any visual components of the Visual Course Planner will not be optional; set colours, fonts, shapes, animations are final.
6. No more than 10 save degree plans can be saved. If a student reaches 10, they must delete already made plans to create another.
7. No email warnings will be sent.
8. The system will not need to handle massive amounts of users at once, maximum ~300.

4.9 ENVIRONMENTAL CONSTRAINTS

1. The system must run on the University of British Columbia Okanagan campus server

other issues? FOIPA

5 FEATURE BREAKDOWN

	Task	Mackenzie	Taylor	Herraj	Jaskaran	Noman
1	**Design Document (total hours)**	10	15	7	7	7
1.1	Create UML Diagram of data model					
1.2	Website wireframes		4			
2	**Backend (total hours)**					
2.1	Setup server	5				
2.2	Create data model	5	5	5	5	5
2.3	Create database					
2.4	Implement Database operations					
2.5	User Authentication					
2.6	Setup CI on server	4				
2.6	Connect CI to GitHub					
3	**User Interface (total hours)**					
3.1	Admin file upload					
3.2	Course list display sidebar					
3.3	Course list filter					
3.4	Login/signup screen					

3.5	Question input					
3.6	Course Drag and Drop					
3.7	Semester/year view and layout					
3.8	DAG Dependency Rearrange	20				
3.9	DAG Dependency Display					
3.10	Plan Favouriting system					
3.11	Plan Notes field					
3.12	Save plan button					
4	**Course Info (total hours)**					
4.1	Academic calendar scraper for all courses			5		
4.2	Course importer from scraper					
5	**MVP Presentation (total hours)**					
6	**User Testing Session (total hours)**					
7	**Degree Plan Optimization (total hours)**					
7.1	Algorithm Design					
7.1	Parameters Input					
7.3	Implement Algorithm					
7.4	Algorithm API					
	Total Hours (all categories)					
	Avg/Week					

6 DURATION

6.1 PROJECT MILESTONES

[Identify the significant project milestones: start date, end date and invoicing dates to the client.]

Start Date	September 25, 2018
Project Charter and Scope	October 16, 2018
Design Document	November 13 ,2018

Minimum Viable Product	January 8, 2019
End Date	April 1, 2019

7 PROJECT BUDGET

Not Applicable.

8 ASSUMPTIONS, CONSTRAINTS AND RISKS

8.1 ASSUMPTIONS

1. Academic calendar is always up to date.
2. System does not have access to UBC course information directly.
3. Verification of a users identity is not required.
4. System will not store confidential student data.
5. Department will supply all necessary degree requirement information.

x data issues – TOS for course data

8.2 CONSTRAINTS

This section identifies any limitation that must be taken into consideration prior to the initiation of the project.

	CONSTRAINT
1	Access to UBC Course Database; unable to receive permission to pull updated course information from UBC Okanagan.
2	Time constraint: 8 months from planning to completion
3	Approval of Scope and Charter. Once approved no additional changes will be supported to design or functionality of the Visual Course Planner.

8.3 RISKS

	RISK	**MITIGATIONS**
1	Project is not completed before the deadline.	Break project into manageable goals, in order to facilitate a clear outline of how much work needs to be done by what date. This will help keep the project on track to completion

2	Team members not contributing as much as needed - other commitments, courses, work etc.	Team members will track contribution time and if discrepancies are noticed, under-contributing team members will either have to contribute more time, or other team members contribute more time.
3	Personal emergencies such as but not limited to: family, and health.	Maintaining proper communication between team members will give us flexibility should emergencies arise.
4	Unable to scrape information from UBC Okanagan due to possibly IT problems preventing the Visual Course Planner being unable to display offered courses for the specific degree selected by user.	
5	UBC Okanagan server crash	

9 PROJECT ORGANIZATION

9.1 ROLES AND RESPONSIBILITIES

NAME	ROLE	RESPONSIBILITIES
Noman Mohammed	Project Manager	Has the accountability for managing the project within the approved constraints of the scope, quality, time, and to deliver the specified requirements, deliverables and customer satisfaction. Also accountable to make sure all team members are on track and can successfully contribute to their other team members and project.
Herraj Luhano	Client Liaison	Maintains communication between the development team and the client.
Jaskaran Lidher	Developer	Works on programming tasks throughout the development of the Visual Course Planner. Can take part in any category of programming.
Mackenzie Salloum	Integration Lead	Manages code repository master branch. Takes the accountability to make sure all code is properly push to the appropriate Git branch including team members branches and merging approved code to the Master branch.
Taylor Siemens	Technical Lead	Manages non-code documents in the repository
Dr. Abdallah Mohamed	Project Sponsor	Responsible for providing direction and support the team. In charge of approving the project scope represented in this document.

9.2 STAKEHOLDERS

Client	Dr. Abdallah Mohamed
Sponsor	Dr. Abdallah Mohamed
Project manager	Noman Mohammad
Project team members	Herraj Luhano, Jaskaran Lidher, Mackenzie Salloum, Noman Mohammad, Taylor Siemens
Other Internal Stakeholders	UBC Students, Dr. Scott Fazackerley, Academic Advisors
Other External Stakeholders	Non UBC students

10 APPROVAL SIGNATURES

The undersigned acknowledge they have reviewed the project charter and authorize the Visual Course Planner project. Changes to this project charter will be coordinated with and approved by the undersigned or their designated representatives.

Dr. Abdallah Mohamed, Project Client	Dr. Abdallah Mohamed, Project Sponsor	Noman Mohammed, Project Manager