Capstone Project Proposal

Submitted to

Computer Science Department

College of Computing Sciences and Engineering

Kuwait University

Advisor

Dr. Maha Al-Alabduljalil

Group Members

Asmaa Sultan 2151117051 {Leader}

Amna Ali 2131110038 {Member}

Sharifa Mousaed 2121111141 {Member}

Thursday, November 28th, 2019

# Project Name:

# Scheduley

## Table of Contents



## Table of Tables

## 1. INTRODUCTION

The last century witnessed a major revolution in technology as it has been present in all aspects of everybody's daily life. Whether in the practical or scientific field, technology has had a major role in their improvement. The situation led to the abandonment of some traditional tools and replaced them with electronic or technical tools thanks to the technology revolution.

While this can be done traditionally, our system will make this process more optimal, smoother, more accurate, and more importantly, faster. That is, save time and effort for all parties involved, allowing them to secure their plans and prepare for the coursework. Problems of the traditional way could include a clueless student who isn't aware of each course's prerequisites and the importance of different courses, meaning some students are unaware that some courses should be taken given the chance, and is better to not leave it to their preferences. Similar to instructors, who may want to offer a certain course that isn't possible for any student with their unique academic progress, to take. Also, a general advisor may not be aware of each student's different plan and need of courses to register in, the number of sections needed, and the instructor's preference to teach them. For example, there exist, several students who do not console with the advisor, therefore the advisor is unaware of those students' plans. And a course offered can be assigned to an instructor that would prefer to teach it at a certain time, rather than task it randomly.

In this project, we will help students and general advisor along with course instructors to plan and schedule courses in the best way, that is, provide both the student and the advisor with a plan for the schedule of the semester, and guiding instructors in regards to what is needed to be taught. Our system will consider each course's prerequisites, its availability, and its priority. It will also help a student visualize and know all the factors involved to help him or her find the best plan for their academic life. Also, the advisor can use this system to set up courses for each semester according to student plans. And finally, instructors can offer courses they'd like to teach while being informed of what courses are needed.

Our project, for now, hopes to reach all students of the Computer Science department at Kuwait University. All students of this department will have a complete log of the sections and courses available from said department in a specific semester, along with a course log that is based on the major sheet for their department, and finally, connected relations between each course and its prerequisite. Leaving no stone unturned for the student to plan with his complete knowledge.

Dr. Maha being the student general advisor herself suggested this idea as a way to deal with the many cases and scenarios she faced with students who came up to her for advising without a clear plan for their academic lives and having very complicated progress.

We implement this as a website, having multiple interfaces and concerns, and show the needed information while hiding complex decision making. Each user has his/her needed information only, to simplify and to secure the process from mistakes or sabotage. Different types of users, such as students and instructors are going to have different information shown. And the manual entry of the courses will be done by tasking an IT member of the computer science department to do so, providing one more user type to our system.

To build our system we will use an incremental model specifically XP [1]. The implementation will be used for both the front-end and back-end web development. For front-end programming, we will use HTML, JavaScript, and PHP. It is also critical to have a database for our system for using the back-end web development to create and manipulate the data as needed.

Some challenges we may face is that our system won't be used by students, hence, the solution for the problem won't be implemented. Therefore, our system will have to be good and pleasing for it to be used. Also, some human decisions could have many more factors than possible to implement in code, some could be personal such as preferences of instructor or time, so our system will also have to be flexible enough for changes and dynamically make new decisions even if some may not seem optimal, logically speaking. Also, we will have to make the system run complex algorithms while appearing to the user as simple as possible, to encourage him/her to use our system for all their academic lives. And finally, to provide integrity of information, at first, our information about the semester schedule and major sheet will need to be manually entered, considering our system is new and not linked to any external system.

Our system will have to be dynamic and personalized for each student, so we must test it for more than one student and hopefully pleasing all. We will need human sources for perfectly testing our system. And using to visualize the behavior of our system to confirm its efficiency. We will also need servers to fetch and store the needed information, along with backup plans and options for these servers.

We will be using the plan for the student as advised plan, the student's course history as academic progress, and the student's general advisor for course planning as a general advisor.

## 2. METHODOLOGY

In this section, we explain the models and tools used to implement our project. In section 2.1, we present the software development model and section 2.2, the tools used during the project implementation.

### 2.1. SOFTWARE DEVELOPMENT MODEL

To implement such a complex system, an ideal model will need to be used. We find an agile process perfect for that. Such a model is better for web-development since it is effective and responsive to change, it also offers effective communication between team members or stakeholders. More specifically we will be using Extreme Programming (XP) model [1]. The reason for choosing this model over other agile process models (such as Scrum [2]) is because it is flexible to change, error, and it saves the time required for project realization, also it encourages pair programming, and finally it provides constant unit testing to every new increment of the system. These are valuable features that improve both the quality and reliability of the system.

## 2.2. REQUIRED TOOLS

For the implementation of the project the following tools will be used to create the source code which is independent of the operating system:

- **Google Docs**: to share the proposal and report documents.
- **Visual Paradigm** or **Lucid Chart**: programs to help us create flow diagrams.
- **cPanel:** Allowed to manage our websites from a web-based interface.
- **TeamGantt:** to do the gantt chart for the plan.
- **JetBrains** or **PHPSTORM**: for the PHP source code.
- **Microsoft Visual Studio:** for HTML source codes, JavaScript source codes, and CSS source codes.
- **Bootstrap** or other frameworks: for creating responsive functionalities.
- **PhpMyAdmin:** to manage the database.
- **MySQL:** to create and manipulate the database and servers. This is done via the database definition language and database manipulation language.
- **GitHub**: a company website that allows the sharing the software development using Git version control.

We will use most of the above tools in parallel with tasks divided among the team members.

# 3. RISKS AND CHALLENGES

In this section, we provide the anticipated challenges and risks we may face to produce the final product, along with how we plan to tackle or reduce their impact. In 3.1 we visit the technical challenges specific to our system and in 3.2 we go through project risks that can keep us from delivering a functional system entirely.

## 3.1. TECHNICAL CHALLENGES

### 3.1.1 Invalid Advised Plan

The system needs to import data of the actual course schedule from servers, hence some advised plans that our system anticipated for the students might not be correct. This requires a real-time modification of the affected advised plans.

### 3.1.2 Finding Optimal Solutions from Limited Options

Students may use the system throughout their academic progress. If their previous choices were unwise based on our system graduation advised plan, the system will try to limit the possible options to help get the plan back to optimum. We also need to consider other scenarios such as when a student fails some courses multiple times, or when two required courses are set at the same slot, or when a course is unavailable due to the sabbatical leave of its instructor. To mitigate the mentioned challenges, we can implement database constraints' exceptions handling such as assertion and triggers. They

can be used for possible typographical errors such as inserting false data, updating irregular data, or deleting critical data.

3.1.3 Hardware Problems

Some computers and servers might crash or suddenly shut down. This will result in a loss of code or important data. For that, we use cloud services to achieve real-time updates.

## 3.2. PROJECT RISKS

3.2.1 Team Issues

Withdraw of a team member during the project may result in an overwhelming effort on the remaining members and reduce the quality of work provided.

3.2.2 Algorithm Complexity and Finding the Optimal Solution

While the system is a hopeful approach it may take longer than the given time to develop a perfect algorithm for the optimal solution. Considering we will need to find the longest path to plan for the entire academic progress. And according to previous research, finding the longest path is an NP-hard problem [3]. Which can only be solved if we use reductions, brute force, or other methods.

3.2.3 Security Issues

Our system might be the prey of external attacks. Due to the complexity of the algorithm and the work required, we might not reach the implement that deals with security attacks.

| Risk | Probability | Impact | RMMM |
|---|---|---|---|
| Team Issues | Medium | Medium | Repartition the whole project to fill the missing |
| Algorithm Complexity | High | High | Heuristic Brute Force |
| Security Issues | Low | High | N/A |

*Table 1: This table shows the risk that the team can faced while doing the WebApp*

## 4. PROJECT DELIVERABLES

This section presents the artifacts delivered by this project. As this project is fully complete, we will deliver the following:

### 4.1 CAPSTONE DOCUMENTS:

- Project proposal: following the CS department guidelines, this proposal marks the first step of our project.
- First report of progress: following the CS department guidelines, the first report of progress will be submitted on February 17th, 2020.
- Second report of progress: following the CS department guidelines, a second report of progress will be submitted on March 16th, 2020.
- Final report of progress: following the CS department guidelines, a final report of progress will be submitted on April 13th, 2020,
- A manual for the student: a readable guideline intended to be delivered during deployment for a student who is new to using our system.
- A manual for the instructor: a readable guideline intended to be delivered during deployment for an instructor who is new to using our system.
- A manual for the general advisor: a readable guideline intended to be delivered during deployment for the student general advisor who is new to using our system.
- A manual for IT maintenance team: a readable guideline intended to be delivered during deployment for the student general advisor who is new to using our system.
- Testing documents: Common errors and mistakes faced during the testing phase.

### 4.2 SOFTWARE COMPONENTS:

- HTML source codes
- CSS source codes
- PHP source codes
- JavaScript source codes
- A WebApp hosted on a server
- MySQL database

## 5. PROJECT PLAN

In the plan section, we describe the plan details, construction, and implementation.

## 5.1. DISTRIBUTION OF TASKS

| N | Task Name | Duration | Predecessors | Assigned To |
|---|---|---|---|---|
| **1** | **Definition & Planning** | 5d | | |
| 1.1 | System Users Permissions Algorithm | 5d | | Asmaa |
| 1.2 | System Security and Login encryption | 5d | | Shareefa |
| **2** | **Design Stage** | 11d | 1 | |
| 2.1 | Interface Design | 11d | | |
| 2.1.1. | Graphic and logo Design | 4d | | Amna, Asmaa |
| 2.1.2. | System Main Page | 2d | 2.1.1. | Amna, Shareefa |
| 2.1.3. | System Sub Page | 5d | 2.1.2. | Amna |
| 2.1.4. | System instructor Pages | 5d | 2.1.1. | Shareefa |
| 2.1.5. | System Students Pages | 5d | 2.1.1. | Asmaa |
| 2.1.6. | System Supervisor Pages | 5d | 2.1.1. | Shareefa |
| 2.1.7. | System IT Member Pages | 5d | 2.1.1. | Asmaa |
| 2.2. | Database Design | 15d | | |
| 2.2.1. | Conceptual View | 5d | | Amna |
| 2.2.2. | Logical View | 5d | 2.2.1. | Shareefa |
| 2.2.3 | Physical View | 5d | | Asmaa |
| **3** | **Implementation Stage** | 19d | | |
| 3.1. | Login Security and Login encryption | 5d | | Amna, Asmaa |
| 3.2. | User's Permissions | 9d | | Asmaa, Shareefa |
| 3.3. | System Main Page | 3d | 3.1. | Amna |
| 3.4. | Courses entry page | 9d | | Amna |
| 3.5. | System instructor Pages | 5d | 2.1.4. | Shareefa, Asmaa |
| 3.6. | System Students Pages | 5d | 2.1.5. | Amna, Shareefa |
| 3.7. | System Supervisor Pages | 5d | 2.1.6. | Shareefa, Asmaa |
| 3.8. | System IT Member Pages | 5d | 2.1.7. | Amna, Asmaa |
| **4** | **Testing Stage** | 30d | | |
| 4.1. | Main Page Testing | 3d | 3.3. | Amna |
| 4.2. | Database Testing | 3d | | Shareefa |
| 4.3. | Configurations Testing | 3d | | Amna |
| 4.4. | Instructor Pages Testing | 3d | 3.5. | Asmaa |

| | | | | |
|---|---|---|---|---|
| 4.5. | Student Pages Testing | 3d | 3.6. | Shareefa |
| 4.6. | Supervisor Pages Testing | 3d | 3.7. | Asmaa |
| 4.7. | IT Member Pages Testing | 3d | 3.8. | Amna |
| 4.8. | Full System Functionality Testing | 7d | | Shareefa, Amna |
| 4.9. | System Permissions Testing | 17d | 3 | Shareefa, Asmaa |

*Table 2: This tables shows the distribution of tasks in our plan*

## 5.2. EXECUTION PLAN (GANTT CHART)

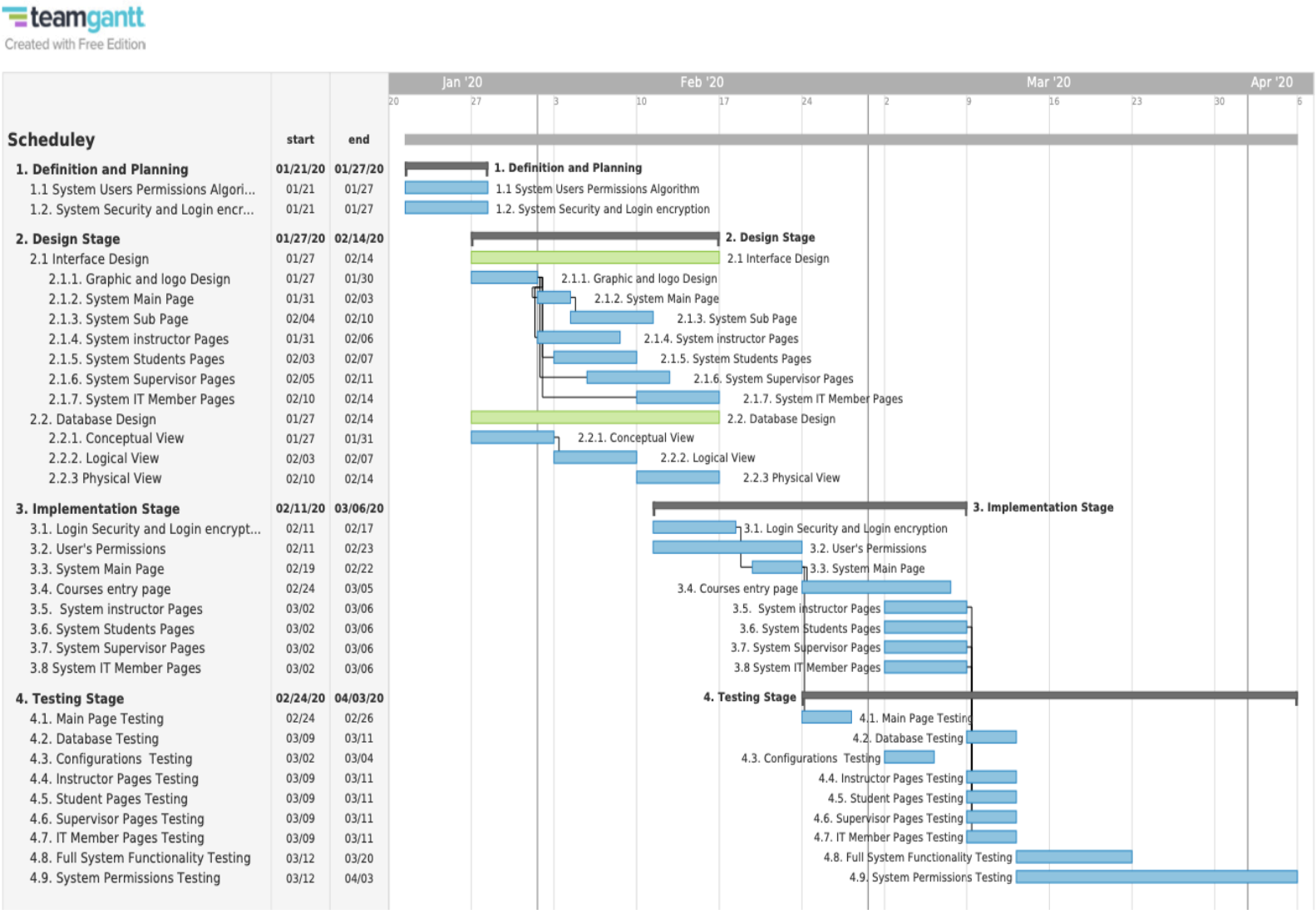


*Figure 1: the plan in the Gantt Chart*

## 6. SUMMARY

In conclusion, this proposal presents a project plan and approach that has the objective of achieving an advising plan system as a web app called Scheduly. It will be developed using an agile process (Extreme Programming). Scheduly will benefit all people in the Computer Science department at Kuwait University. The work on our project will take place in the spring semester of 2019. The web-application design will be developed, and team members will start implementing the necessary parts, leaving time to test our system or any delay.

## 7. REFERENCES


(1)
Wells, B. D. (n.d.). *Extreme Programming: A gentle introduction.* Retrieved from http://www.extremeprogramming.org/index.html

(2) Ken Schwaber, J. S. (n.d.). *WHAT IS SCRUM?* Retrieved from https://www.scrum.org/resources/what-is-scrum

(3)
Aristotle, T. S. (n.d.). NP-Hardness. In T. S. Aristotle, *The Perfect Critic.*