Some unclear points that I want to clear before starting the assignment:

Since the author is the owner or me. Then all the plan that I created last week; I will highlight it boldly. But I will highlight only an important word(mostly topic), not the whole sentence, it won’t look good(The topic will be bold black and description will be normal. At some point if I would like to add my thoughts than I will change the font color to red). And talking about the “red and italic” I would only use this for all of the application parts.

Waterfall Technique: The waterfall technique is a more linear approach than other software development methodology. It goes in a sequence from one step to the other. Since only one phase happens at a time it is one of the easy to follow and manage technique in software development.

**Scenario:** I the author of this document and the owner of the companyoversees creating an application for the university registration system. We will have a team of 19 people including 12 software engineers. Our assignment for this project is to merge the BYUI grad plan with the registration system. We need to find an optimum schedule that will decide the number of sections needed for any specific class in that semester.

**Meetings:**

**Project Discussion Meetings: Our first meeting will be understanding, planning and drafting meeting. First, we will understand what the project is about and we will try to predict the time, resources, and profit that can be made from this project.**

* + Since we are a small team of 19 people, we want all the people but custodian to attain this meeting.
  + As said earlier the main agenda of this meeting is to assign everyone their role and try to find the optimum process for doing this project. We will also divide people into different groups.
  + We will look at the nature of the project. We will come up with different checkpoints and smalls goals that will guide us to our main goal without losing track. The checkpoint will be created for the completion of requirement writing and prototyping, design documentation and testing, analysis and program design, coding and testing, and final product. We will try to get as much as information about the project if needed we will talk to Peter and Patricia. Since the database already exists. We will divide our team into three pieces of front-end, back-end, and implementer.

Technically it will be only two team front-team and back-end team; the implementer team will be made out of few representatives in front-end and few representatives from the back-end.

We will also decide that the meeting of the whole team, but custodial and secretary will be held twice every week on Monday and Wednesday. And the meeting of an individual team will be in the first 15 minutes of the workday. The meeting time can be expanded if the plan for the whole day does not come. The meeting of implementer will be after half time for 15+ mins every day.

* + This overall project discussion meeting will be only once and will not be over until everyone is satisfied.

**Sunday-Wednesday Meeting: This is the meeting that happens every Sunday and Wednesday at the start of the day. After this meeting the regular team meeting will happen. This meeting time is not exact and can go any long or short, but for average it will take around an hour.**

* Everyone except custodian and secretary needs to attain this meeting.
* The agenda of this meeting will be to see if both teams are on right phase. Some changes that have come along or will come will be discussed. The technical or any other difficulty will be discussed. The checkpoint and the work done so far will be discussed. Any changes needed will be discussed.
* We will make sure everyone is satisfied with each other work. If some-one wants any changes in the project then they will be given chance. The comments and opinion of the customer will be addressed. There will be group conversation will the representative of the university so all the engineers know directly what they want.
* As said, this meeting will be twice a week.
* **Daily-Team Meeting: This is the daily meeting of team members.** 
  + All the team members will attain it.
  + To come up with the work plan for a day.
  + Every worker will have knowledge that what they are doing for the day. And at the end of the day what they should have in order to show for next day meeting. All team members need to discuss what they accomplished in last day of work.
  + This meeting will be held every day.

**Implementer meeting:**

* + Two people from each team will be selected for this role and meeting.
  + Make sure that front-end and back-end are linked together, there is no information leaking, and all the needed information are sent from UI to server and vice versa.
  + If there is any problem in UI because of the way back-end people programmed or vice versa and the problem requires changes in requirement and documentation, then those problems must be taken back to the team and also should be discussed in Monday-Wednesday meeting.
  + This meeting will be every day after break for 15 mins

**Documentation:** The first document we will write will be the requirement, the second document will be on how those features will be implemented, and the third will be the instruction on using those features

* The main author and editor of all three documents will be a technical writer. The reviewer and second author will be Rochak. But all the engineers and designers must contribute to the first and second documentation. The final documentation will be done only by Technical writer.
* *The audience of the first document will be a software engineer and UX designer who will design the features and the customer will be the school. The audience of our second document will the software engineer as well as a maintenance engineer. And the audience of the third document will be the user of our software.*
* *The document would be very helpful for the engineer to note what feature they have implemented and how they are going to do it. I think the biggest help will be to maintenance engineer later.*
* The requirement writing will take 10% of the total time, instruction will be done at the end of the project, and documentation of features and implementation will be done in first 10%-20% time period.

**Roles:**

* Project manager: Project manager will be Rochak: the owner of the company*. To be the project manager one mush have a good management background and business-minded. Should be able to take hard and difficult decision when needed.* Rochak must do all the regular talking with the customer as well as present our product to our customer. *Rochak should be good at managing his workers.*
* Custodian: As regular custodian will be Chrissy. She will clean the office in the evening.
* Secretaries: Stan and Sally will be secretary and will help Rochak for office work. They will be very friendly and help to fill the bridge between Rochak and his worker.
* Technical Writer: Teri will be a technical writer and Rochak will be the reviewer. *One should have spent good enough time in tech-industry to be in this role. These people will write all the documentation need for software.*
* Designer and basic front-end engineer: Ursula, Xavier, and Ingrid will be a designer and HTML coder. *To be in this position one needs to have a good technical background and sense of style.* They will design the front page of the software.
* Software engineer: These people will plan the whole software. They will architect the implementation and working of the software. Abe, Britney, and Jack will be a software engineer*. To get these roles one need to have a good knowledge of software engineering as well as coding.*
* Developer: Claire, Emily, Grace, and Larry are a software developer. They will write code under the design and direction of software engineers. *To be in this role one needs to be good at programming. Their job is to write code.*
* Tester and Debugger: Frank and Doug will be our tester and debugger. Frank will be writing our test case and Doug will try to fix bugs. *To be in this role one need to have experience as a QA engineer.*
* QA engineer: Holly and Keith are QA engineer. They will be trained by Abe and Britney so they do their QA job. They will check the quality of the software produced. They will have to report to Frank and Dough about any issues they find.

**We are diving these people in two-team of front-end and Backend. People in front-end and backend are listed below, if you think you are placed in the wrong place you can discuss it in the project discussion meeting.**

**Front-End:** Ursula, Xavier, Claire, Emily, Grace, Larry, Keith

**Back-end**: Abe, Britney, Ingrid, Frank, Doug, Holly

**Implementer-team**: Ursula, Claire, Abe, Ingrid

**Checkpoints:**

**These are the checkpoint and the number in percent given is the time period they will be done in the project.**

1. What is the name of the checkpoint? The name of the checkpoint is listed in the list of the checkpoint itself.
2. How long (roughly) do you expect it to take to reach this checkpoint? We are not looking for a detailed effort estimation such as what we do in CS 416, but rather a ballpark estimate. Also, if this checkpoint is to occur more than once, mention it here. The checkpoint dates are given in percent beside the name of checkpoint. It is in percent because the total project time will come form the discussion in the project discussion meeting on the very first day.
3. How will you know that the checkpoint is reached? This is sometimes called the "exit criteria." For checkpoint 1: Requirement writing and prototyping must be completed. For checkpoint 2: Implementation and testing checkpoint must be completed by 25% of project time. Analysis and program design must be completed by 40% of total time of project. Coding and testing must be completed by 80% of the total time of the project. And Final product is to be made by 80% of the total project time.
4. **Requirement writing and prototyping checkpoint (0- 10%)**
5. **Implementation and testing checkpoint (10%-25%)**
6. **Analysis and Program Design checkpoint (25-40%)**
7. **Coding and testing checkpoint (40%-80%)**
8. **The final product (80%)**

*If something very difficult to solve or some legal issues come in between, or if any extension needed then 20% of the total maximum time allocated can be added in the project time. But by putting final product in 80% owner of this company wants to show that they can build the product 20% faster than their competitors.*

Reflection:

The team is small and so the project is. Everything is planned quite good and there is testing during the beginning phase to make sure we find our error before coding and implementation. It can be tough if you found an error in the final testing phase and we have to go back and change our requirement, but if that case doesn’t come than this method can be time-saving, easy to follow and better than any software methodology.   
But, if in case something really hard to solve thigs appears or if the customer is not satisfied with something, we might have to spend little more time in comparison to following of other software methodology.

Overall, there is a danger as well as benefits in this methodology. Looking at the project size and team size I feel like there should not be any problem in communication and making changes for customer satisfaction. The implementation and documentation phase have testing which will stop lots of error before even coding which means saving a lot of time. I think, personally I would like to follow this methodology because of the small team size and small project size.

Resources:   
[1]. W. Royce, "Managing the Development of Large Software Systems," *Proceedings of the IEEE WESCON*, pp. 328-338, Aug. 1970   
[Online] Available: [http://dl.acm.org/citation.cfm?id=41801](https://content.byui.edu/file/fb36352f-44a4-473d-bb81-1e5a2ce36646/1/CS%20432%20PDFs/Royce_waterfall.pdf)

[2]. D. Dischave, "A Waterfall Systems Development Methodology... Seriously?," Global Enterprise Technology, Sep. 2012  
[Online] Available: [http://get.syr.edu/news\_alt.aspx?recid=401](https://content.byui.edu/file/fb36352f-44a4-473d-bb81-1e5a2ce36646/1/02%20-%20Dischave%20-%20A%20Waterfall%20Systems%20Development%20Methodology.pdf)

[3]. A. P. Merce, “Waterfall Model: What Is It and When Should You Use It?,” *Airbrake Blog*, 02-Nov-2017. [Online]. Available: https://airbrake.io/blog/sdlc/waterfall-model. [Accessed: 26-Sep-2019].

[4]. J. Periyasamy, “What is Waterfall model and list its advantages, disadvantages,” *What is Waterfall model and list its advantages, disadvantages*. [Online]. Available: http://jobsandnewstoday.blogspot.com/2013/04/what-is-waterfall-model.html. [Accessed: 29-Sep-2019].

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Exceptional 100% | Good 90% | Acceptable 70% | Developing 50% | Missing 0% |
| Accuracy 40% | It is completely obvious which development methodology is described. Any knowledgeable person would be able to identify the methodology based on this document. | There is nothing to add and nothing wrong; the development methodology is completely described. One part of the plan may be misclassified as**bold** or *red/italic* | There exists one small problem (factual error or missing component). | There exists one large or multiple small problems (factual errors or missing components). | Large parts of the development methodology are inaccurately described or missing. |
| Application 30% | It is obvious that real thought went into the application (*the red/italic part*) of the plan. | The development methodology is applied to the scenario in an uncontrived way. | Every aspect of the scenario is incorporated into the development methodology. | Large parts of the plan are overly vague, do not appear to be related to the scenario, or do not appear to be related to the development methodology. | No attempt was made to apply the development methodology to the scenario. |
| Reflection 20% | The reflection cuts to the heart of the strengths and weaknesses of the development methodology. | The strengths and weakness of the development methodology are clearly communicated. | One strength and one weakness is mentioned in the reflection. | Little thought or effort was put in the reflection part of the paper. | The reflection part of the paper is missing. |
| Professionalism 10% | The paper is easy to read and ideas are clearly communicated. | Everything is properly cited, there are no grammar or spelling errors, and writing style is "professional." | One instance of a spelling error, grammar error, incomplete citation, overly verbose, poor formatting, or poor writing. | A citation is missing where one is needed (plagiarism alert!). | Gross spelling/grammar errors or other aspects of the writing that make the paper difficult to read. |

**Claim**: I claim 100% on this assignment.

**Evidence:**   
Have spent more than 12 hours total this week for this class.  
Instructions can be made a lot better, but have done my best to fulfill the requirement.  
I would suggest and ask to grade the paper with some comment so we could know what can be done to make it better and not repeat the same mistake.