

Introduction, Purpose, and Team

The purpose of this proposal and project overall is to create a degree progress portal for students to better access and understand how far to completion they are. University leadership has requested this and the student body as a whole stands to benefit from having this information readily available to them.

Documented in page 4 under “Project Description” in the charter details the end goals of the project and how to achieve it. Such as a portion of the web portal that details through graphical interface and the security necessary to display aforementioned information securely. The section also outlines what determines success such as being defect free, within deadline, and being functional.

Leading the charge on this project will be Jane Smith. Our project sponsor. This role is largely managerial. Handling the interactions between roles, giving the approval of changes and being informed of all happenings within the project. The importance of this role is to maintain vision of the project as a whole to all parties involved. Most importantly the financing of the project.

Assigned tasks:

- Supervise

Our project manager is John Doe. The project manager oversees the project at a closer level. Being there to assure quality assurance and documenting numerous reports about the project. The importance of this role to the tasks below is straight forward. Without someone competent to compile reports and assure quality the project as a whole will suffer.

Assigned tasks:

- Management and budget plans
- Create gantt chart
- Create charter
- Monitor project
- Final project report
- Project reflections
- Manage deliverables

Arthur Brown is our core stakeholder. That's who will be facilitating between the team as a whole, and other stakeholders and third parties alongside the project sponsor. He is to be informed of various milestones upon their completion so as to relay this information to those who are indirectly involved with the project.

Assigned tasks:

- Facilitate communication

Shila Cole is the most experienced of the team members working on this project. Team members actually construct and build what the charter and WBS outline. Shila is accountable for her peers and aids in various aspects of the project. She tests and assures quality in the code and interfaces created by her peers.

Assigned tasks:

- create test plan for system components
- test scenarios defined in test plan
- perform integration testing with front and back ends for communication.

Ana Fischer is another team member. She is responsible for large portion of the hard coding and user interfacing for this project. Her role is important because without it there may not be anyone competent enough to accomplish the tasks required.

Assigned tasks:

- | | |
|--|--|
| • Develop UI mockup | • Perform integrate integration testing |
| • Architectural design mockup for user | with front and back ends for |
| • Develop GUI | communication. |
| • Develop web code to receive back-end | • Transition web application to production |
| data to be placed within GUI | environment(go life) |
| • Test scenarios defined in test plan | • Assess and manage risks throughout life |
| | cycle. |

John Jones is the final team member for this project. This team member will be responsible for more back-end data construction. This role is important as the back-end is where a large portion of problems may occur and so requires a knowledgeable figure to interact with it.

Assigned tasks:

- Create architectural design for back-end
- Develop back-end code to process data and send it to web application
- Test scenarios defined in test plan
- Perform integrate integration testing with front and back ends for communication.
- Transition web application to production environment(go live)
- Assess and manage risks throughout life cycle.

The Constraint and Me

I have opted to be “John Jones” during this exercise, the back end system analyst. This role is important because without it the whole project can't function. This role is tasked with creating communication functions to allow the database to communicate with the web GUI being developed by Ana.

The tasks mentioned above and more are listed on the gantt chart attached. These organizational predecessors are required before John Jones can go off to design the back-end architecture because without this scope and method to approach the project one can't know what's expected of them. The back end design is required before he can actually build it because without a design how will he know what he's building? The process would be sluggish and

riddled with time consuming errors. This role aids in performing tests and transitioning the project to a live setting

In light of receiving additional funding management has requested a callout box be implemented into the website side of the project. This extra feature requires little in way of extra time commitment. An additional two days is expected to implement this feature. One day for the back end functionality and a second day to properly create the front end and link the two. Thorough testing to functionality should be done during the allocated testing section of the gantt chart.

This modification is acceptable within regard to the triple constraint matrix in that the additional money allows for the addition of another feature that does not significantly alter the project in any way. The outline given during the project briefing was an additional week of time to implement the feature. With these additional liberties this addition is acceptable.

A Major Risk

One of the highest probability risks is that of a separate project being built at the same time, and the two not interacting well once they are both rolled out and live. One could cause the other bugs and crashes due to them not being constructed with the other having existed when development started.

Deploying to production is an issue as old as software where-in the rollout to live requires the modifying of live and active files. The wrong edit could cause persistent errors without a notable source, or even break the entire system. These two dilemmas together create a recipe for an easy disaster and should be accounted for.

As noted the chance for error for carelessly modifying the live site for a project rollout is high. The number of dependencies. Stated in the WBS the entirety of the 1.4 section “controlling” is at risk of an error during 1.3.4 “transition to production”. Regardless of how well a project is tested and managed, unless it's tested against other projects happening at the same time the interaction will not be smooth and could lead to frantic stress to isolate and control the problem, while the product/project is live.

Implications

The potential fallout extends beyond developer scrambling. Stakeholders will be upset and begin to doubt the professionalism and credibility of the team that made it and the business as a whole. The time estimation for the project could need to be pushed back several days or more to restore order. Thus causing scheduled future projects to be delayed and throwing the entire business into disarray.

The Tact of Mitigation

There are a few respectable ways to mitigate this risk and by association the others it causes. Tackling the issue of improper compatibility between concurrent projects first the steps needed are to collaborate with them. Set time to establish meetings between milestone phases and deliver prototype builds or notifications of what aspects of the system are being modified to alert the other team that this could be an issue. Noting what files are being changed and how. Furthermore at the final testing phase before rollout the two projects should have one final collaborate internal test where-in the two projects see if they are compatible. While the two

projects will need more time overall to completion, the added benefit to the other two aspects of the triple constraint are maintained as well as public image. It goes without saying a late, perfect, project is better than a rushed broken one.

A Challenger Appears

The other problem of deploying to production is the human aspect. While the testing above could go flawlessly it's pointless should the human element of the project be unable to replicate the actions of the successful test perfectly. Fixing this is as simple as creating a script during testing to deploy the project. This script will execute the same thing, each time. If it was successful through repeated testing then it will be successful when deploying to live so long as there are no other unexpected changes to the live server. Finally, after creating the successful script package the application or change immediately. This ensures what the script does, will be the interacted with properly. There is no cost to these methods as far as the triple constraint is concerned, and should already be best practice when deploying projects.

One Last Note

A final note, to minimize mass hysteria, is the usage of feature flags. Feature flags allow an update to be pushed, but invisible unless enabled. The coding is there, but not turned on. This feature allows it to be deployed and allow for one final bug test. The biggest example is this is Facebook. Where in they update constantly but are never offline to do so.

Conclusion

To reiterate and conclude. The needs of the project is a working degree completion checker for the university. One potential pitfall is a second project being constructed at the same time and interfering with the source code, causing out projects to error with each other. Communication with the other project and collaboration are recommended for this.

A second pitfall is the human aspect of deployment. Human error is a factor in all things and mitigating risks of this nature should be striven for in all tasks. The usage of scripting the launch and feature flags help here.

Ultimately the take-away is so long as potential risks are communicated between concurrent projects, the triple constraint maintained, and the rollout process as automated as possible there will be no project fatal issues and a smooth launch is more than probable.