Project Management Plan

For: The University of Texas at Dallas

Project: Campus Parking Availability App

By: Mohammad Asif, Dylan Capece, Everett McCrea, Luke Padgett, Blake Stroud

SE 4381.002 - Spring 2020

**Table of Contents**

Business Landscape......................................................................................................... 2

Customer Responsibilities............................................................................................. 2

Assumptions and Dependencies.......................................................................... 2

Acceptance Criteria............................................................................................. 3

Scope............................................................................................................................... 4

Components......................................................................................................... 4

Out of Scope........................................................................................................ 4

Work Breakdown Structure............................................................................................. 6

Stakeholders Mappings................................................................................................... 12

Communication Plan....................................................................................................... 13

Governance Plan............................................................................................................. 14

Risk Management Plan................................................................................................... 15

Change Management Plan.............................................................................................. 16

Lessons Learned List...................................................................................................... 17

**Business Landscape**

**Customer Responsibilities**

* The University of Texas at Dallas (UTD) assigns a point of contact (POC) for the Development team to coordinate their efforts with and address existing and emerging concerns.
* UTD facilitates the requirement elicitation process for development team by giving them free access to the necessary facilities and persons
* UTD determines the type of technology it wants to use for tracking empty space.
* UTD provides the hardware needed for the project.
* UTD surveys and determines where the required hardware be installed.
* UTD install the required hardware in the designated locations.
* As UTD enforces a color-coded parking permit, the project requires UTD to do any necessary rearrangements in its open-area parking spaces.
* UTD makes its open-area parking spaces controllable based on its permit policy.
* UTD compensates the development team, as agreed, upon the delivery of the product.
* UTD pays the cost of labor, equipment, and services associated with the project.
* In case of taking any fees for the service provided by the App, UTD integrates and normalizes the app with its current parking billing system or any other DBs.
* UTD facilitates and cooperates in the possible software-hardware integration pilot testing
* UTD participates in the last integration testing.
* UTD is responsible for providing an updated map of the various parking lots around campus and the surrounding roads leading in and out of them.
* UTD shall provide timely feedback about malfunctions and defects in software as well as make design changes to the front-end system should they deem it necessary.

**Assumptions and Dependencies**

* Users will need to provide their license plate number and parking tag color to when first using the app.
* KLY Inc. shall use industry best practices when building the hardware, establishing the wide area network, and developing a mobile application to reduce the number of defects in the final deliverable.
* KLY Inc. shall use industry best practices when handling delicate user information so ensure the security of our users personal information.
* Users are expected to make their best attempt at securing their personal account information.
* UTD shall choose which parking lots the hardware for this project will be installed at.
* The mobile application shall display the most recent geographic layout of the UTD campus to its users.
* Should the UTD campus change the layout of its parking lots, UTD shall provide KLY Inc. an updated map of the campus.
* Should any hardware components be damaged, UTD shall report the damages to KLY Inc. at their earliest convenience.
* Data collected (user information, application downloads, parking statistics, etc.) by the system will be provided to UTD should they send a formal request to KLY Inc..
* Users will have to provide their UTD credentials to have access to the mobile application.
* Hardware upgrades and replacements shall only be conducted should both UTD and KLY Inc. agree to the arrangement.
* Individuals without a parking tag and/or valid parking license number shall not be allowed to use the mobile application.

**Acceptance Criteria**Acceptance of Services:

* The development team shall notify the University of Texas at Dallas upon the completion of the services by sending a notice of completion. Upon receiving this notice, the University of Texas at Dallas will have seven days to examine the acceptability of the delivered services and their conformance to the requirements as stated in the SOW. Should the delivered products be found underperforming and/or unacceptable from what was agreed upon in the SOW, the University of Texas at Dallas shall send a notice of revision to the development team.

The development team shall deem the services as accepted and finished if any of the following occurs:

1. Seven days have passed since the notice of completion has been sent with no response of non-conformance from the University of Texas at Dallas.
2. The University of Texas at Dallas notifies the development team of their acceptance of the deliverables provided.
3. The University of Texas at Dallas utilizes any of the delivered products in any shape or form of business operations other than testing the deliverables for conformance.

**Scope**

The purpose of this project is to help UTD students and other affected parties determine which parking lots have available spots on campus. KLY Inc. is delivering a system comprised of a mobile application, parking lot gate hardware, and a host computer system in order to show the levels of various parking lots around campus in real-time. The mobile app allows users to filter which parking lots have available spaces based on colored parking permits and filter out parking lots that do not have the desired colored spaces.

This project was created in order to reduce the stress of finding a parking spot on the UTD campus as well as save students’ time attempting to find a parking space. KLY Inc. has decided to develop an app that is able to display the number of open spots at any given parking lot on the UTD campus. The mobile application would utilize user-friendly design and have a relatively low learning curve for users who may not be as skilled with mobile technology. This mobile application will greatly alleviate one the biggest problems on the UTD campus and make efficient use of people's time and money. With this app, KLY Inc. is dedicated to solving a real and present issue that commuting students and faculty have to face every work day when arriving on campus.

**Components**

1. Ability to view the various parking lots on the UTD campus and see the parking availability of each one.
2. Ability to communicate to the user which parking lots are full and which have valid parking still available.
3. Ability to filter which parking spaces are displayed to fit the user’s needs.
4. Ability to update the mobile app map display with real-time data that is captured as cars drive into various parking lots.

**Out of Scope**

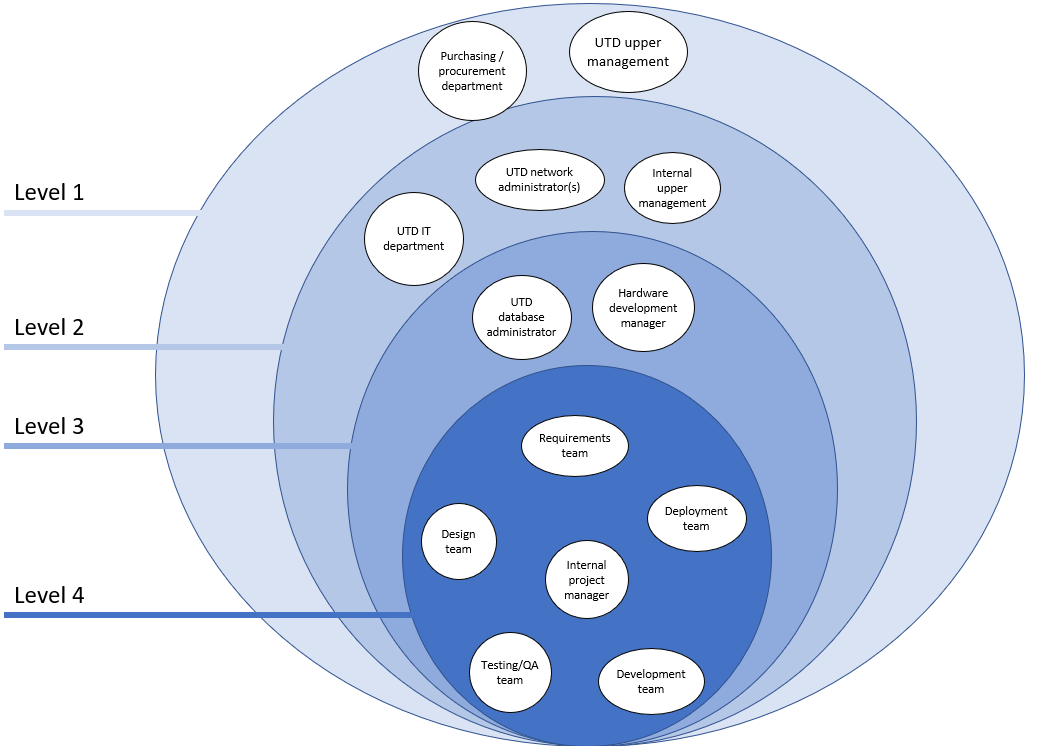
* The mobile application will not work with any other universities or areas other than UTD.
* The system will not be able to detect when parking lots have closed down or are under maintenance.
* The mobile application will not support phones that do not have access to the Apple App Store or the Google Play Store.
* The mobile application will not advise users approximations/estimates on when various parking lots experience heavy traffic.
* The mobile application will not be able to determine the exact geographic location of the available spaces, only that they are within a specific parking lot.
* The mobile application will not enforce any UTD parking rules/laws and will not issue tickets/violations to offending users.
* The mobile application will not validate which parking spot the user has parked in - it will only assume that the user has parked in the highest tier colored parking space that they have access to.

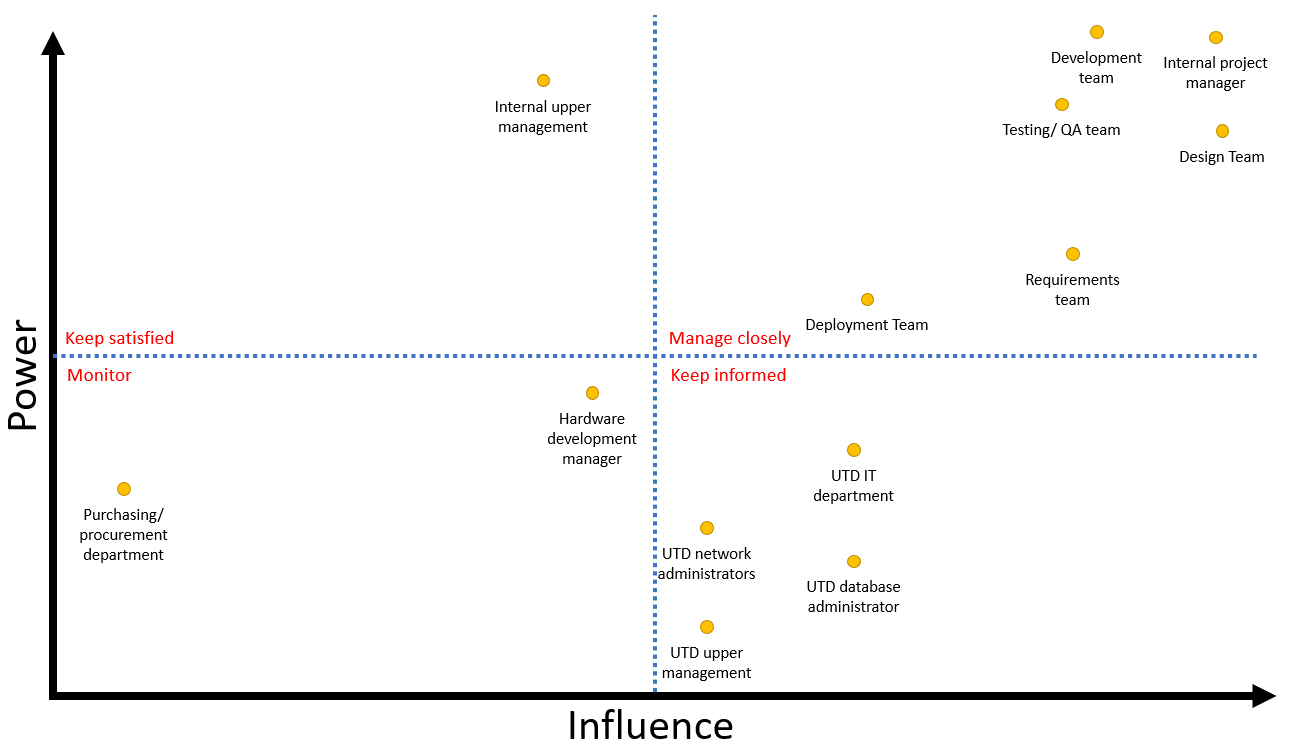
**Work Breakdown Structure**



|  |
| --- |
| **Project** |
| **Requirement** |
| Create stakeholder map |
| Create stakeholder analysis |
| **Host Computer System Requirements** |
| Draft network requirement questions |
| Draft real-time updating requirements questions |
| Draft OS requirements questions |
| Draft data encryption requirements questions |
| Review all host computer system requirements questions |
| Finalize all host computer system requirements questions |
| **Server/back-end** |
| Draft DB schema requirements questions |
| Draft DB security requirements questions |
| Draft DB applications requirements questions |
| Draft DBMS requirements questions |
| Review all server/back-end requirements questions |
| Finalize all server/back-end requirements questions |
| **Web page** |
| Draft login/logout page requirements questions |
| Draft payment page requirements questions |
| Draft user authentication requirements questions |
| Draft payment validation requirement questions |
| Draft admin page requirements questions |
| Review all web interface requirements questions |
| Finalize all web interface requirements questions |
| **Mobile application** |
| Draft UTD map display requirements questions |
| Draft payment Page requirements questions |
| Draft user authentication requirements questions |
| Draft payment validation requirement questions |
| Draft real-time updating requirements questions |
| Draft login/logout page requirements questions |
| Review all mobile application requirements questions |
| Finalize all mobile application requirements questions |
| **UI/UX** |
| Draft UI/UX mobile application UI/UX requirements questions |
| Draft UI/UX web page UI/UX requirements questions |
| Review UI/UX requirements questions |
| finalize UI/UX requirements questions |
| Hold interview with customer |
| Create project requirements document |
| Review project requirements document |
| **Finalize project requirements document (milestone M01)** |
| **Design** |
| **Host computer system design** |
| Network connections design (App& Hardware) |
| Real-time information updating design |
| Host computer system data encryption design |
| Host computer system OS integration design |
| Review host computer system design |
| Finalize host computer system design |
| **Server/back-end design** |
| Design schema |
| Design CRUD applications |
| Design DB security systems |
| Design DBMS integration with system |
| Review server/back-end design |
| Finalize server/back-end design |
| **Mobile application** |
| Design UTD map display |
| Design real-time updating system |
| Design payment page |
| Payment validation design |
| Design login/logout pages |
| User authentication design |
| Review mobile application design |
| Finalize mobile application design |
| **Web page** |
| Design payment page |
| Payment validation design |
| Design login/logout pages |
| Design admin page |
| User authentication design |
| Review web page design |
| Finalize web page design |
| **UI/UX** |
| Design mobile application UI/UX |
| Design web page UI/UX |
| Review UI/UX design |
| Finalize UI/UX design |
| Draft final design document |
| Review design document |
| **Finalize design document (milestone 02)** |
| **Development** |
| **UI/UX** |
| Develop mobile application UI/UX |
| Develop web page UI/UX |
| Review/Finalize UI/UX development |
| **Host computer system** |
| Build network between mobile app & RF sensors |
| Build real-time updating system |
| Install OS |
| Develop data encryption |
| Review/Finalize Host computer system development |
| **Server/back-end** |
| Integrate DBMS into system |
| Build schema |
| Build service tier applications |
| Develop security protocols to protect DB data |
| Review/Finalize server/back-end development |
| **Mobile application** |
| Program UTD campus map display |
| Program real-time updating map display |
| Program login/logout pages |
| Program payment page |
| Program transaction validator |
| Program user authentication procedure |
| Review/Finalize mobile application development |
| **Web page** |
| Program login/logout pages |
| Program payment pages |
| Program admin page |
| Program payment validator |
| Program user authentication procedure |
| Review/Finalize web page development |
| Draft final development document |
| Review development document |
| **Finalize development (milestone 03)** |
| **Testing** |
| Write Unit test |
| Review Unit tests |
| Finalize unit tests |
| **Test mobile application** |
| Test UTD map display |
| Test real-time parking lot updating |
| Test login/logout pages |
| Test user authentication systems |
| Test payment page |
| Test payment validation system |
| Finalize mobile application testing |
| **Test Server/back-end** |
| Test DBMS integration |
| Test DB security |
| Test DB schema |
| Test service tier applications |
| Finalize server/back-end testing |
| **Test host computer system** |
| Test networks between mobile app and RF sensors |
| Test real-time parking lot updating |
| Test installed OS |
| Test system data encryption |
| Finalize host computer system testing |
| **Test Web page** |
| Test login/logout pages |
| Test payment pages |
| Test admin page |
| Test payment validation system |
| Test user authentication system |
| Finalize web page testing |
| **UI/UX** |
| Test mobile application UI/UX |
| Test web page UI/UX |
| Finalize UI/UX testing |
| Defect testing |
| Regression testing |
| Conduct end-to-end testing |
| Draft final testing document |
| Review testing document |
| **Finalize testing (milestone M04)** |
| **Deployment** |
| Setup host computer system |
| Begin capturing car information from parking lot gates |
| Launch database |
| Establish connection between UTD database and app database |
| Open network ports/sockets |
| Connect system components |
| **Launch app** |
| Release on Apple App Store |
| Release on Google Play Store |
| Train client on system |
| Draft final deployment document |
| Review deployment document |
| **Finalize deployment (milestone M05)** |

**Stakeholders Mappings**





**Internal project manager:** KLY Inc. project manager responsible for keeping project on budget, schedule, and scope. Responsible for drafting the Statement of Work and for discussing project matters with KLY Inc. upper management. Internal employees should meet with him face-to-face or send an email to discuss topics relating to the project. External parties should send an email.

**Design Team:** Team in charge of creating and enforcing the design patterns and blueprints that shall be used during the development of the UTD Parking Availability App Project. Design Engineer Lead should be contacted via email first before establishing any face-to-face meeting.

**Development team:** Team in charge of creating the codebase for which the UTD Parking Availability App will run off of. This team is responsible for both front-end and back-end of the project while following the design laid out by design team. Development Engineer Lead can either reached via email or voicemail.

**Testing/ QA team:** Team responsible for testing the codebase created by the development team. Determines which functionalities are working as intended and which are not. Reports back to development team each week with bug findings. Testing Engineer 12 should be contacted first via phone call or email before going to Testing Engineer Lead.

**Requirements team:** Team that collects the requirements from the customer (UTD) and interprets the requirements for the design, development, testing, and deployment teams. Customers should contact Requirements Engineer Lead via phone call. Internal employees may meet with Requirements Engineer Lead any weekday after 6:00 PM at her office.

**Deployment Team:** Team responsible for connecting all the developed systems together and delivering final project to customer. Deployment team members should work closely with customer’s IT department and database administrator to ensure seamless integration of system into customer’s technological infrastructure. Deployment Team Lead should be reached via phone call on Tuesdays and Wednesdays, and email every other weekday.

**UTD IT department:** IT department at UTD responsible for ensuring maintenance of UTD systems and services. UTD IT department should be made aware of any decisions involving the technology used by the system. Contact either the UTD IT help desk or send email to IT department representative Carl Smith at email cfm150430@utdallas.edu.

**UTD database administrator:** Administrator of the UTD student records database. Determines which applications will be allowed access to the UTD database and has full autonomy over changes made to student records schema design. Communication should be limited to phone calls, video calls, and emails only.

**UTD network administrators:** Manager of the UTD network and determines which devices/systems are permitted access to UTD bandwidth. Is available to meet each Friday from 3:00 to 7:00 PM.

**UTD upper management:** Primary sponsors of the UTD Parking Availability App Project. Upper management includes University president, dean of undergraduate learning, dean of graduate learning, and president of bursar office. Communication limited to meetings that agree with everyone’s schedule.

**Internal upper management:** KLY Inc. upper management and direct supervisor of internal project manager. Responsible for funding project. Can be reached by emailing his assistant any time during the work week.

**Hardware development manager:** Manager responsible for ensuring the development of the gate infrastructure goes along smoothly. Will communicate with the internal project teams to pass along information about how the gate infrastructure functions and how it should be use to develop the app. Available for meeting every Monday, Tuesday, and Friday form 12:00 – 2:30 PM.

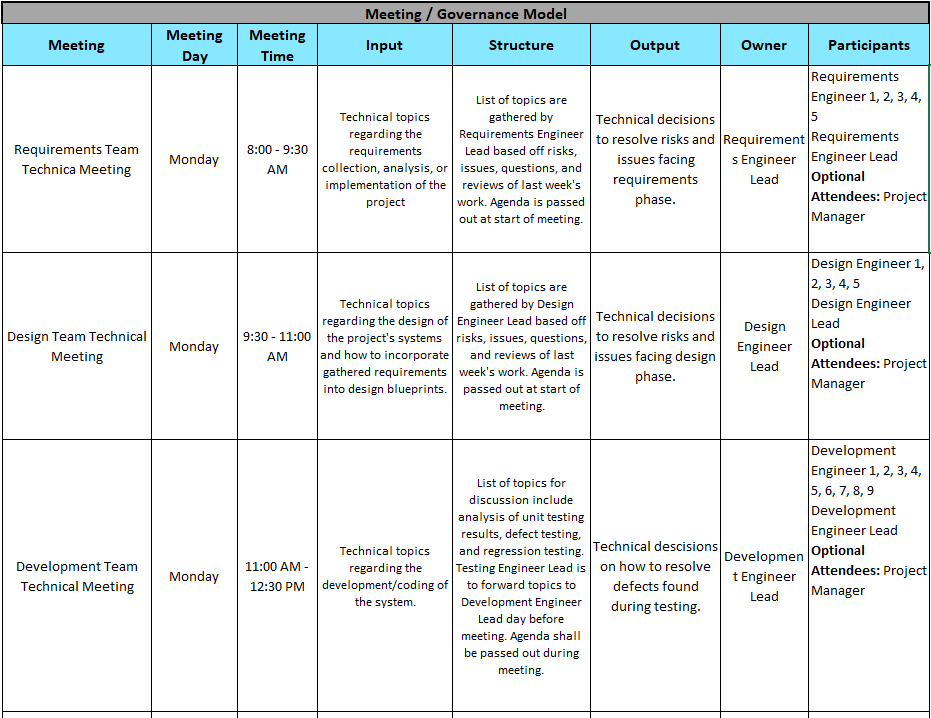
**Purchasing/ procurement department:** Internal department that is responsible for placing all orders on required equipment for KLY Inc. projects. Should resources need to be purchased for a project, representative must be contacted at least one month in advance. Can set up a meeting with department representative any time during the weekdays.

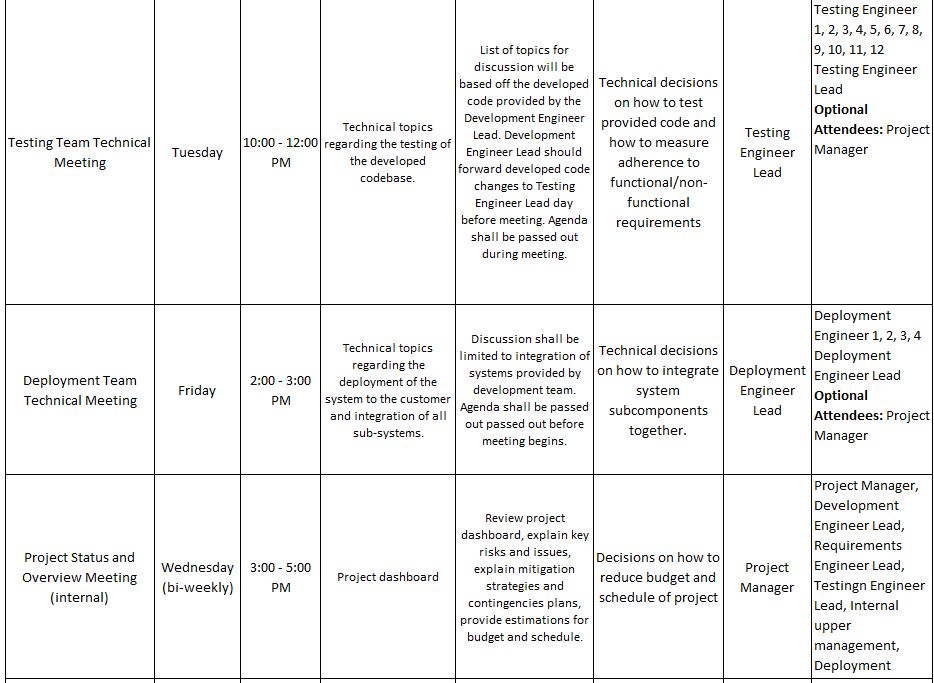
**Communication Plan**

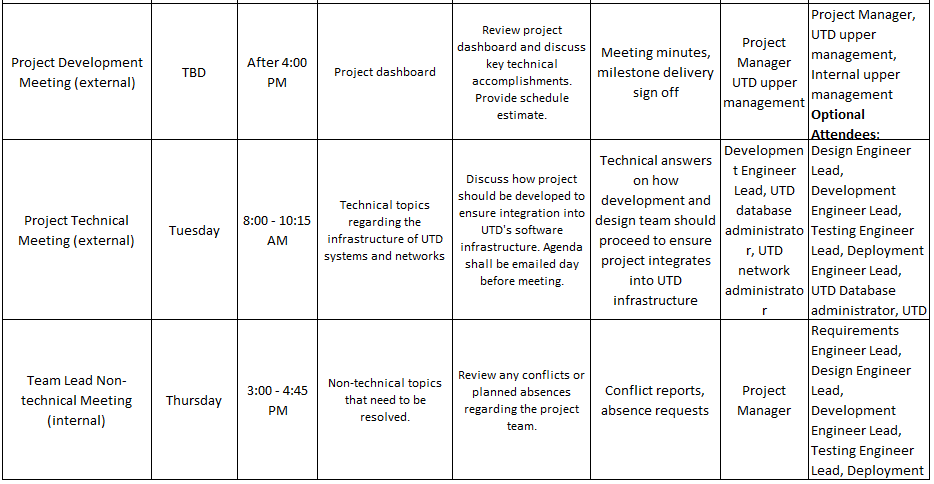
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Stakeholder** | **Method of communication** | **Days Available** | **Times** | **Subjects** | **Description** |
| **Internal project manager** | email, meeting | M | W | F | Sa | 6:00 AM – 5: 00 PM | Risks, issues, conflicts, estimations, change requests, project updates, schedule forecast, | Project manager is point of contact for all external contacts and customers. All customer queries and project status updates should be directed to project manager first. |
| **Design Team** | email, meeting | M | Tu | W | Th | F | 8:00 AM – 5:00 PM | Mitigation strategies, risks, issues, schedule changes, resource allocation, meeting times | Design Engineer Lead is point of contact for entire design team. Should information received be intended for specific design team member, Design Engineer Lead shall inform/forward that information to the employee. |
| **Development team** | email, voicemail, meeting | M | Tu | W | Th | F | 8:00 AM – 5:00 PM | Mitigation strategies, risks, issues, schedule changes, resource allocation, meeting times | Development Engineer Lead is point of contact for the entire development team. Information regarding bug fixes should be provided each week. Should information received be intended for specific design team member, Design Engineer Lead shall inform/forward that information to the employee. |
| **Testing/ QA team** | email, phone call | M | Tu | W | Th | F | 8:00 AM – 5:00 PM | Mitigation strategies, risks, issues, schedule changes, resource allocation, meeting times | Testing Engineer Lead is point of contact for the entire development team. Development Engineer Lead shall contact testing team with bug fixes made each week. Should information received be intended for specific design team member, Design Engineer Lead shall inform/forward that information to the employee. |
| **Requirements team** | meeting, email, phone call | M | Tu | W | Th | 10:00 AM – 4:00 PM | Mitigation strategies, risks, issues, schedule changes, resource allocation, meeting times | Requirements Engineer Lead is point of contact for entire requirements team. Any changes in requirements or customer interviews should be communicated as soon as possible. Should information received be intended for specific design team member, Design Engineer Lead shall inform/forward that information to the employee. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Deployment Team** | email, meeting, phone call | M Tu W |Th Sa | 8:30 AM – 3:30 PM | Mitigation strategies, risks, issues, schedule changes, resource allocation, meeting times | Deployment Engineer Lead is point of contact for deployment team. Any issues with installation dates or customer meetings should be reported immediately. Should information received be intended for specific design team member, Design Engineer Lead shall inform/forward that information to the employee. |
| **UTD IT department** | Email, meeting | M Tu W Th F Sa Su | 9:00 AM – 4:00 PM | Technical queries, meeting times | Representative from the UTD IT department is selected as the point of contact for the project team to communicate with (cfm150430@utdallas.edu). Should any questions regarding the architecture or the UTD systems be needed, they should be forwarded here. |
| **UTD database administrator** | email | M Tu W F | 9:30 AM – 4:30 PM | Technical queries, schema design questions | Key figure in determining what systems have access to UTD student information database. Development and deployment teams should communicate with DBA whenever information about the structure of the database is needed. |
| **UTD network administrators** | Email, video meeting | M Tu W Th F | 9:30 AM – 4:30 PM | Technical questions, port information | Key figure in determining what systems have access to the UTD networks. Development and deployment teams should establish communication whenever information about network design is needed. |
| **UTD upper management** | Meeting, email, video meeting, phone call | M W Sa Su | 12:00 AM – 3:00 PM or by appointment | Project requirements questions, schedule questions, meeting times, project status reports | People in charge of paying for the project. Information regarding changes in overall project delivery schedule should be communicated to this party as soon as possible. |
| **Internal upper management** | Email, voicemail | ThF Sa | 1:00 PM – 2:00 PM or by appointment | Meeting times, changes in schedule, budget impacts, issues, changes in scope, project status reports. | Supervisor of entire project. Responsible for allocating funds to the project and is key factor in determining which projects are taken on. Project manager should communicate major issues with the project to the internal upper management when they arise. |
| **Hardware development manager** | Meeting, phone call | M Tu Th F | 3:00 PM – 6:00 PM | Meeting times, project development reports, technical queries | Responsible for overseeing the development of gate hardware. Communication should be established between Development Team to ensure that codebase is developed to work properly with chosen hardware. |
| **Purchasing/ procurement department** | Meeting | W Th F Sa | 7:00 AM – 5:00 PM | Meeting times, purchase requests | Party responsible for obtaining necessary equipment needed to complete project tasks. Meeting should be established early in project life cycle to ensure timely delivery of equipment. |

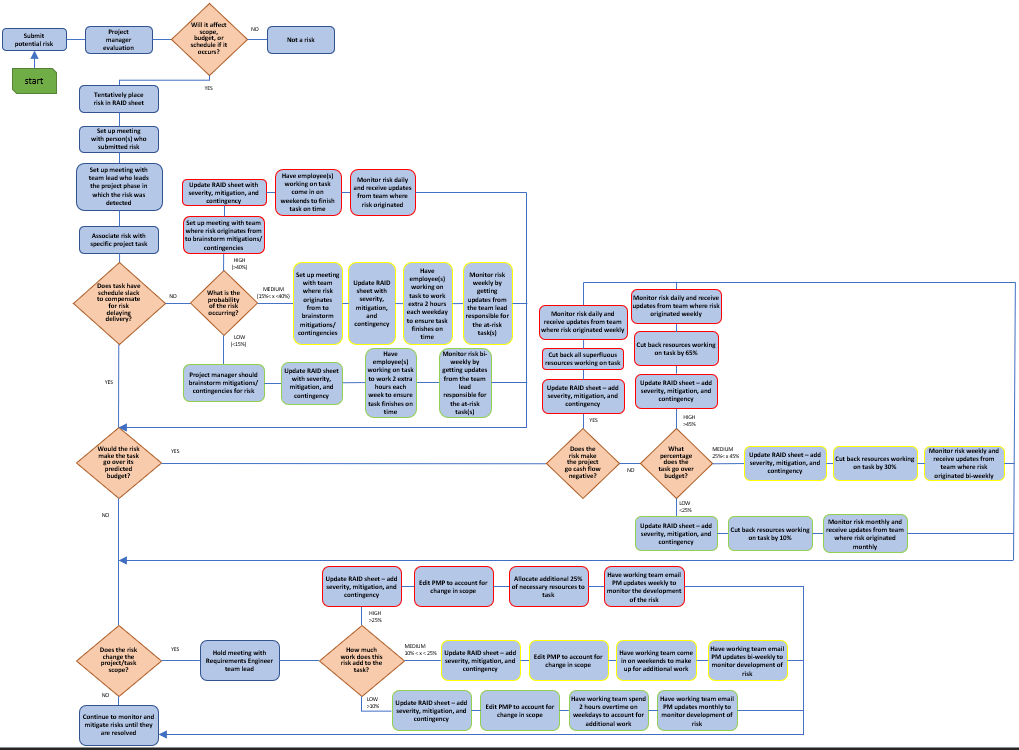
**Governance Plan**







**Risk Management Plan**



|  |  |
| --- | --- |
| **Flow Step** | **Flow Step** |
| Submit potential risk | A risk is generated from a team member or project manager. Risk is forwarded to the project manager for further evaluation |
| Project manager evaulation | Project manager will look over the submitted risk and evaluate it to determine if the risk poses a threat to changing either the scope, budget, or schedule of the project overall |
| Will it affect scope, budget, or schedule if it occurs? | Project manager must determine if the submitted risk has the chance of changing the scope, budget, or schedule of the project in any way if it were to take place. |
| Not a risk | The project manager has determined that the submitted risk cannot/will not affect the scope, budget, or schedule of the project in any form. If there scope, budget, or schedule is not affected, than the submitted risk is not actually a risk. |
| Tentatively place risk in RAID sheet | List the risk in the RAID sheet but do not give it a severity rating yet. Project manager will gather more information about the risk and its owner(s) before filling out RAID sheet |
| Set up meeting with person(s) who submitted risk | Talk to the person who originally submitted the risk to get more details about the risk. In this meeting, get the following information: issues leading to risk creation, probability of risk occuring, cost impact of risk, schedule impact of risk, and scope impact of risk. |
| Set up meeting with team lead who leads the project phase in which the risk was detected | Meeting with the appropriate team lead should include talking about the nature of the risk (schedule risk, budget risk, scope risk) and going over what strategies are being used to counteract the risk. Team lead should provide an updated probability of the risk occuring, risk's impact on the budget, and scope change percentage. |
| Associate risk with specific project task | Specify which task in the project schedule that the risk originates from and update RAID sheet with information |
| Does task have schedule slack to compensate for risk delaying delivery? | Project manager should determine whether or not the task that the risk is associated with is on the critical path according to the project schedule. |
| Set up meeting with affected team to brainstorm solutions | Get the team who is responsible for the at-risk task together and come up with mitigation plans that will resolve the risk before it becomes an issue. |
| Update RAID sheet with severity, mitigation, and contingency | Project manager will go back to RAID sheet and fill out remaining information that was not gathered the first time. |
| Initiate mitigation plan to prevent risk from turning into issue | Put the mitigation plan into affect immediately in order to prevent risk from happening. Inform affected people of this decision. |
| Monitor risk daily and receive updates from team where risk originated | Ensure that the risk is not turning into an issue by keeping an eye on how the at-risk task is developing. Team that risk is affecting should email project manager weekly with status updates regarding the risk until the risk has been fully mitigated. |
| Set up meeting with affected team to brainstorm solutions | Get the team who is responsible for the at-risk task together and come up with mitigation plans that will resolve the risk before it becomes an issue. |
| Update RAID sheet with severity, mitigation, and contingency | Project manager will go back to RAID sheet and fill out remaining information that was not gathered the first time. |
| Monitor risk weekly by getting updates from the team lead responsible for the at-risk task(s) | Ensure risk is being monitored and kept under controll by checking up with the team lead for which ever phase the at-risk task is in (i.e. development team lead if the at-risk task is related to the development of the project). A regular 30+ minute talk should be had where the team lead explains what he/she is doing to mitigate the risk. |
| Project manager should brainstorm mitigations/ contingencies for risk | Project manager should forumlate solutions to risk that will mitigate the risk without putting another task at risk. |
| Update RAID sheet with severity, mitigation, and contingency | Project manager will go back to RAID sheet and fill out remaining information that was not gathered the first time. |
| Monitor risk bi-weekly by getting updates from the team lead responsible for the at-risk task(s) | Ensure that the risk is being monitored and does not become a greater risk. Updates received from team lead should include a status of the at-risk task. |
| Would the risk make the task go over its predicted budget? | Project manager should look at the estimated budget in their project plan to determine if the at-risk task in going over that predicted amount. |
| Does the risk make the project go cash flow negative? | Project manager should look calculate the cost of other tasks happening concurrently to at-risk task as well as how much is expected to be paid for the delivery of the milestone to determine if the at-risk task sends the project to a negative cash flow. |
| Cut back on superfluous resources working on task | Assign all non-essential resources on at-risk task to other tasks that need to be worked on. |
| Update RAID sheet – add severity, mitigation, and contingency | Record risk on the RAID sheet and add its severity (high), mitigation plan, and contingency plan. |
| Monitor risk weekly and receive updates from team where risk originated | Make sure employee(s) working on at-risk task update project manager weekly about the budget of the task. Project manager should use these costs forecast the project budget and make adjustments to the project plan. |
| How much does the task go over its assigned budget? | Determine a numerical impact to the project budget that the risk would cause if it turned into an issue. |
| Update RAID sheet – add severity, mitigation, and contingency | Project manager will update the RAID sheet so that the risk has the appropriate severity associated with it (high) as well as a mitigation and contingency plan. |
| Cut back resources working on task by 50% | Take away 65% of the resources currently working on the assignment and allocate them to other tasks. |
| Monitor risk daily and receive updates from team where risk originated weekly | Project manager should check the risk daily to make sure task does not go over budget. The team working on the task should provide detailed breakdown of what part of the budget was spent to accomplish the task. |
| Update RAID sheet – add severity, mitigation, and contingency | Project manager will update the RAID sheet so that the risk has the appropriate severity associated with it (medium) as well as a mitigation and contingency plan. |
| Cut back resources working on task by 30% | Take away 30% of the resources currently working on the assignment and allocate them to other tasks. |
| Monitor risk weekly and receive updates from team where risk originated bi-weekly | Project manager should check risk weekly to ensure it does not turn into an issue. The team working on the task should provide detailed breakdown of what part of the budget was spent to accomplish the task. |
| Update RAID sheet – add severity, mitigation, and contingency | Project manager will update the RAID sheet so that the risk has the appropriate severity associated with it (low) as well as a mitigation and contingency plan. |
| Cut back resources working on task by 10% | Take away 10% of the resources currently working on the assignment and allocate them to other tasks. |
| Monitor risk monthly and receive updates from team where risk originated monthly | Project manager should check risk monthly to ensure that it is mitigated before moving on to new tasks. The team working on the task should provide detailed breakdown of what part of the budget was spent to accomplish the task. |
| Does the risk change the project/task scope? | With the information provided from both the person who submitted the risk and the team lead who has jurisdiction over the task in which the risk was spotted, decide if the risk changes any part of the project or task's scope. |
| Hold meeting with Requirements Engineer team lead | Discuss risk further with Requirements Engineer Lead to determine what impact on the project's/task's scope this risk has. In this meeting, determine how much of a change in scope percentage this risk poses. |
| How much work does this risk add to the task? | Project manager should determine what percentage of work the risk may add to the at-risk task based upon the information provided by the Requirements Engineer Lead. |
| Update RAID sheet – add severity, mitigation, and contingency | Project manager will update the RAID sheet so that the risk has the appropriate severity associated with it (high) as well as a mitigation and contingency plan. |
| Edit PMP to account for change in scope | Project manager will report change in scope on the PMP so that its permanently recorded and available for reference. |
| Allocate additional 25% of necessary resources to task | Ensure that the additional workload is met with additional resources to ensure task is completed on time. |
| Have working team email PM updates weekly to monitor the development of the risk | Make sure that team responsible for working on at-risk task keeps project manager up to date with developments with the task. |
| Update RAID sheet – add severity, mitigation, and contingency | Project manager will update the RAID sheet so that the risk has the appropriate severity associated with it (medium) as well as a mitigation and contingency plan. |
| Edit PMP to account for change in scope | Project manager will report change in scope on the PMP so that its permanently recorded and available for reference. |
| Have working team come in on weekends to make up for additional work | Have the working team come in on weekends until all additional work is completed so that resources are not taken away from other tasks. |
| Have working team email PM updates bi-weekly to monitor development of risk | Make sure that team responsible for working on at-risk task keeps project manager up to date with developments with the task. |
| Update RAID sheet – add severity, mitigation, and contingency | Project manager will update the RAID sheet so that the risk has the appropriate severity associated with it (low) as well as a mitigation and contingency plan |
| Edit PMP to account for change in scope | Project manager will report change in scope on the PMP so that its permanently recorded and available for reference. |
| Have working team spend 2 hours overtime on weekdays to account for additional work | Working team should spend additional hours working on at-risk task so that schedule is maintained and delivery of project is not delayed |
| Have working team email PM updates monthly to monitor development of risk | Make sure that team responsible for working on at-risk task keeps project manager up to date with developments with the task. |
| Continue to monitor and mitigate risks until they are resolved | All risks should be continued to monitored and actively mitigated in order to ensure that they do not turn into issues and affect the scope, budget, or schedule of the overall project. |

**Change Management Plan**

**Lessons Learned List**