CMPT 371 – Team 3

Risk Management Plan

Version History

Update for Incremental Deliverable 4:

Rating Change:

- Illness:
  - Probability: Significant -> Severe
  - Impact: Moderate -> Significant
- Holidays
  - Impact: Minor -> Significant
- Headset Damage
  - Probability: Minor -> Moderate
- Motion Sickness
  - Probability: Moderate -> Significant
- Eye Strain
  - Probability: Minor -> Moderate
- Physical Injuries
  - Probability: Minor -> Moderate

Description Change:

Increasing the probability of the VR risks because of user testing.

Risk Change:

- Add a new risk "Schedule".
  - Probability: Significant   Impact: Significant   Priority: Significant
- Add a new risk "Implementation".
  - Probability: Significant   Impact: Significant   Priority: Moderate
- Add a new risk "Automated Testing".
  - Probability: Moderate   Impact: Moderate   Priority: Significant
- Add a new risk "Time Estimation".
  - Probability: Severe   Impact: Moderate   Priority: Significant

Top 10 Change:

- Update the position and information on list.
  - Add Time Estimation, Schedule, Implementation.
  - Remove VR Controls, DICOM, SIIM Innovation Challenge.

Update for Incremental Deliverable 3:

Rating Change:

- o Eye Strain:
    - Priority: Moderate -> Significant
- o DICOM
    - Probability: Moderate -> Minor

Risk Change:

- o Add a new risk "Oculus Package".
    - Probability: Significant   Impact: Significant   Priority: Significant
- o Add a new risk "SIIM Innovation Challenge".
    - Probability: Moderate   Impact: Significant   Priority: Significant

Top 10 Change:

- o Update the position and information on list.
    - Add Oculus Package, SIIM Innovation Challenge.
    - Remove Unity License, Data Corruption.

Update for Incremental Deliverable 2:

Rating Change:

- Holidays:
  - Probability: Minor -> Moderate
- Illness:
  - Probability: Moderate -> Significant
- Due Date:
  - Priority: Significant -> Severe
- Client Leaving
  - Impact: Significant -> Severe

Description Change:

- Color is from blue to red because of colour blindness.
- Grammar fix.
- Update the introduction to explain a bit more for some instructions.
- Update headset damage for contingency plan.

Risk Change:

- Add a new risk "Bugs".
  - Probability: Significant   Impact: Significant   Priority: Significant
- Add a new risk "DICOM"
  - Probability: Moderate   Impact: Significant   Priority: Significant
- Add a new risk "VR Control"
  - Probability: Moderate   Impact: Moderate   Priority: Moderate

Top 10 Change:

- Update the position and information on list.
  - Add Bugs, DICOM, and VR Control.
  - Remove Physical Injuries, Continuous Integration, and Client Leaving.

Table of Contents

Introduction

The Rating scale:

Negligible (0% - 20%)

Minor (20% – 40%)

Moderate (40% - 60%)

Significant (60% – 80%)

Severe (80 - 100)

Instruction:

Probability – Probability rating of the risk to occur.

Impact – Impact rating of the risk to the project.

Priority – Priority of the risk to our project.

Risk – Description of what the risk this.

Project Impact – Potential impact of the risk to the project.

Consequences – Consequences that can happen to the project if the risk does happen, and activate the contingency plan.

Response – Response strategy to prevent the risk from causing issues.

Contingency Plan – Risk Contingency Plan to deal with the risk if it happens.

Insurance – Plan in place that would lower the cost of the risk to the project.

Virtual Reality Risks

Motion Sickness

Probability:

Significant

Impact:

Moderate

Priority:

Significant

Risk: Motion sickness occurs because of slow refresh rate, poor resolution, and visual vestibular mismatch (vestibular system is part of the inner ear that monitors movement and helps control balance).

Project Impact: Motion sickness can turn people off of VR, which could affect the entire project if the problem becomes too significant.

Consequences: Motion sickness can cause users to experience general discomfort, nausea, headache, disorientation and fatigue.

Response: Less virtual movement and extensive testing for different type people. Try different strategies to reduce discomfort, for example adding a nose, glasses, body, to the user. Add a session timer.

Contingency plan: Equalize the sensory cues by fixating on an object far away.

Insurance: Before testing, have the tester answer a questionnaire telling them there is a possibility of motion sickness and give their signature for approval.

Eye Strain

Probability:

Moderate

Impact:

Significant

Priority:

Significant

Risk: Too much exposure to HEV (high energy light) for long period of time. Staring at the screen that is inches from your eyes. Distortion in the picture.

Project Impact: Eye strain could make the software unusable by a number of affected users.

Consequences: Eye strain can cause permanent damage to the retina, and causes short-sightedness, and nausea from distortion.

Response: Significant focus on steady FPS (frame rate per second) to avoid distortion, and unnatural strain on the eye. Add a session timer. Add an option for the user to adjust the focus for each eye camera, or couple of pre-set view.

Contingency plan: Because the symptom is physical damage, there is not much the developer can do, except doing their best to help the user prevent the risk from happening.

Insurance: Have the tester answer a questionnaire telling them there is a possibility of eye strain and get their signature for approval.

Physical Injuries

Probability:

Moderate

Impact:

Significant

Priority:

Moderate

Risk: Physical Injuries in VR can happen from bad frame rate, no visual representation of the user's arm in VR, or uneven or messy surroundings.

Project Impact: Physical safety is an important concern for the public, so the developer would have to put in a significant amount of time to perfect it.

Consequences: Motion sickness would give the user physical pain from falling, colliding with hard object, breaking bones, just to name a few. It can also cause damage to household items like the computer screen, water cups, or desk.

Response: Give the user the freedom of customizing their workstation to match their real-life environment, and design an intuitive and easy to use UI (user interface). Doing tests with users to help with the design of the UI.

Contingency plan: Because the symptom is physical damage, there is not much the developer can do, except doing their best to help the user prevent the risk from happening.

Insurance: Add the possibility of physical injuries into the User Guide and advise the user to prepare the work space before using VR.

Headset Damage

Probability:

Moderate

Impact:

Significant

Priority:

Moderate

Risk: Accidental drops occur during programming and user testing session, or when the user accidentally drops the VR headset.

Project impact: The impact would be significant for the team because if an accident does happen to occur then it would greatly diminish our ability to test and program, delaying the due date.

Consequences: Most of the consequence to the team would be the time cost, and spending cost for the owner.

Response: Always have member of the team on site to provide assistance for the tester, and leave most of the physical dealing with the VR head set to the team member.

Contingency plan: Because the symptom is physical damage, there is not much the team can do, except to do their best to preventing risk from happen. We can move our work to the second VR work station that will be provided by the stakeholder.

Insurance: Have the tester answer an questionnaire, and tell them that there is a possibility of accidentally breaking to the VR headset, so only the team would be allowed to put the headset on and take it of, and If any accident does occur we would have to follow the guideline. Ask the tester to give their signature for approval.

VR Controls

Probability:

Moderate

Impact:

Moderate

Priority:

Moderate

Risk: The stakeholder wants the team to create the program using keyboard as the main control. The user can't see the keyboard or their hand, so using keyboard is not a good choose of control for VR.

Project impact: It would create more work to figure out a way to get keyboard control to work smoothly with the VR.

Consequences: More time spent trying to figure out the control rather then working on the main project.

Response: Find alternative controls that would work better for VR and convince the stakeholder to change his mind about keyboard control.

Contingency plan: Make the VR work with the keyboard as best as we can.

Insurance: Find ways to minimizing the use of keyboard, and better solution to replace it.

Program Risks

Unity License

Probability:

Moderate

Impact:

Significant

Priority:

Significant

Risk: There are four types of Unity license: Personal, Education, Plus, and Pro. The big difference between the license is the ability to publish the project. The Personal, and higher all have the ability to publish, but the Education doesn't. Accidental edits between Educational and Personal versions could impact the project metadata.

Project Impact: The impact to the project would be huge. If any of the team member accidentally edited in the wrong version without knowing, the mistake could jeopardize the entire project.

Consequences: The main problem would be the project will lose its ability to be published.

Response: Try not to open the project on the university's computer, do most of the work on your own personal computer, and be cautious of the risk.

Contingency plan: Try to identity when the problem might have started, and look through the past save files to see if there is a version before it. Check for the amount of data missing and with the help of the problem project update the older version back to the current states.

Insurance: Consistently doing backup of the project, check 3 project version, a one day old, a 3 day old and a week old version, to try and cover all possibilities.

Continuous Integration

Probability:

Minor

Impact:

Moderate

Priority:

Minor

Risk: Releasing Code with errors, or testing functions still inside, because building the program is automatic with continuous integration. Nobody is fixing the broken code, because it is easier to find bugs then fixing it.

Project impact: Delay programmable time, or risk the team by releasing an unpolished code.

Consequences: Bad response from the client when trying an unpolished program.

Response: Keep the master build to be releasable at all times, the development branch to do merges, and tests. Only if the scripts have passed all tests is it allowed to commit to the master branch. If any test fails, then the code has to be fixed and do all the tests until finished. Make a list of bugs that appears and the date of when they appeared, and the date of when they are fixed.

Contingency plan: Roll back the git, or use the backup to get back to working as soon as possible.

Insurance: Create a backup of the Github, in case anything went wrong.

Data Corruption

Probability:

Minor

Impact:

Significant

Priority:

Moderate

Risk: Files become suddenly inoperable or unusable. Computer loses power or crashes during saves. Saving files on bad section of your hard drive, or other storage media, or writing wrong information to the files.

Project impact: Depending on the item corrupted, the lost time and data can be massive.

Consequences: Information inside of the file is written at the wrong place, data becomes scrambled, leaving the file physical unreadable. Programs or operating system can also develop corrupted files. The corrupted files may not be able to open.

Response: Do not remove the corrupted files right away, try to recover the data using software like Recuva, Puran, or Disk Drill. The program may help recover some lost or damage files. Save often, test often, and wait for the save to finish before shutting down the program.

Contingency plan: Use the backup files and try to recover the files with the software.

Insurance: Make backup of the file you're working on, two or three versions. If the project is large, keep a few copies at different location, such as USB flash drives or different computer. Run anti-virus scans regularly.

Server Crash

Probability:

Minor

Impact:

Significant

Priority:

Minor

Risk: The school server can crash unexpectedly, causing the computer which is connected to stop working.

Project impact: Stops the work flow of the project, may also impact communication and program development significantly.

Consequences: Halting the development process for the Oculus Rift, delaying deliverables.

Response: Prepare a backup way of communication and programming options.

Contingency plan: Because the symptom is a problem outside of our control, there is not much the team can do, except to doing their best to prevent a crash from having too much impact to the project.

Insurance: The leaders would have to try and get in contact with the team member and assign task that doesn't require the use of VR.

Communication

Probability:

Minor

Impact:

Significant

Priority:

Minor

Risk: The Slack server can go down.

Project impact: The team would lose all communication, cause the team to go blind.

Consequences: No communication between team members, because Slack is the only tool that all the team members share.

Response: Project manager acquires all the team member's contact information for immediate transfer.

Contingency plan: Project manager contacts all team member to transfer to the backup to an already established communication tool.

Insurance: Create a backup communication tool that team can use when the main tool is down. Backup all communication.

Bugs

Probability:

Significant

Impact:

Significant

Priority:

Significant

Risk: Errors, flaws, glitches in the program that produce incorrect or unexpected results. There are five broad categories of bugs, token error that occurs when using word that is not in the programming language, syntax error occurs when using incorrect grammar, syntax constraint error occurs when finding out that it can't legally carry out one of our instructions after executing, execution error happens when executing a program, and intent errors are errors that did not get detect by the compiler or runtime check system.

Project impact: Can cost the whole program if an important bug gets released with the final version of the program. Lots of time will be spent discovering bugs., adding more work to the project.

Consequences: Unhappy stakeholder if released program with bugs, lots of time will spend fixing bugs.

Response: Bug party to try and find bugs, try to fix bugs as soon as bugs have been found, and make note of all bugs.

Contingency plan: Bug party after an important bug is found, because fixing that bug might cause many other bugs to appear.

Insurance: Create backup file often, and add all bugs that have happened into the testing plan, so it can check if the bug came back.

DICOM

Probability:

Minor

Impact:

Significant

Priority:

Significant

Risk: Not getting a DICOM decoder for our program. The DICOM decoder is a very important part of our project, it is our only way of getting the image and data out of the DICOM files.

Project impact: It is one of the must-haves for our project, so not having it affects the functionality of the project.

Consequences: Not being able to get the patient data, and patient's image.

Response: Try to talk with the stakeholder and figure out a way to get one of the open source DICOM decoder/viewer online.

Contingency plan: Make our own DICOM decoder/viewer.

Insurance: Learning about DICOM and how the images and data might be stored, and ways to extract the information.

Oculus Package

Probability:

Significant

Impact:

Significant

Priority:

Significant

Risk: The Oculus package is creating many bugs for our project, and it gets updated quite frequently which will affect us depends on how much it changes. The Oculus package is very important to our project. It is our only way of connecting the Oculus to Unity.

Project impact: The package is a must-have for our project, and it is the only one available as of now. We have no choice but to deal with its many flaws.

Consequences: Not being able to use Oculus Rift with our program. It would cost the entire project if the Oculus package failed to work.

Response: Try and learn as much possible about the Oculus package, record all bugs that appear to have any connection with the Oculus package, and how it got fixed.

Contingency plan: The Oculus package is created by Oculus themselves, so there isn't much we can do except filing bug reports, and hope Oculus won't release a broken package.

Insurance: Learning about Oculus package, and how it is integrated to our project.

Implementation

Probability:

Significant

Impact:

Significant

Priority:

Moderate

Risk: Fail to finish everything on the task list. Not implementing everything we planned, or required. Unfinished or buggy implementation.

Project impact: Fail to meet the task on the requirement document, not living up to the expectation for us. Fail to finish the task for one ID, pushes the problem onto the next ID, which just increases the work.

Consequences: Unhappy and unsatisfied shareholder, and unhappy team members.

Response: Try to implement as much as possible, and take the time that is required for the task in to consideration, to see if you need more hands on the task.

Contingency plan: The group will follow the Triage team's plan to see if any implementation is going to be removed for this ID, or which one to focus on.

Insurance: Have a full schedule of all the task that needs to get done, to help the team member be organized and mindful of what will take place.

Automated Testing

Probability:

Moderate

Impact:

Moderate

Priority:

Significant

Risk: Not getting automated testing for our programs. All the testing that is going to be done to the program, will be manual.

Project impact: More work on testing, and put extra importance on find out bugs in the program, would have to spend more time on testing.

Consequences: Added work for testing, and increased chance of bugs getting past.

Response: Focus more time on getting test working, and checking for bugs.

Contingency plan: Create detail testing plan for manual testing.

Insurance: Find out which part of the program is prone to bugs, and record all bugs and errors that have been found.

Group Risks

Illness

Probability:

Severe

Impact:

Significant

Priority:

Moderate

Risk: Team members catching the flu or other illnesses.

Project impact: The impact depends on the response of the team, if the team keeps up with the response plan, then the impact can be negligible. It will also depend on how long the sickness will last.

Consequences: The only major problem to the project is losing a team member for the duration of the sickness.

Response: The best way to help lower the impact of the risk is to have everyone know what each person is doing. For that to happen require multiple pair program and code review or all the team member, we can add a request page for pair programming.

Contingency plan: Because the symptom is physical damage, there is not much the team can do, except to do their best to prevent the risk from having too much impact to the project.

Insurance: The team leader would have to ensure that pair programing and code reviews are taking place.

Holidays

Probability:

Moderate

Impact:

Significant

Priority:

Moderate

Risk: Holiday is a common occurring event, and is something that the group can expect. The risk is only when they team member is traveling somewhere far away or doing something involve high risk that the risk would occur.

Project impact: The impact is minor for normal holidays, but It might grow when holidays become closer to the due date.

Consequences: The only major problem with the holidays is unscheduled holidays, and response plan not being met.

Response: The best way to help lower the impact of the holidays is to have a code review or a pair programming session, for the people that will be hard to get in contact with during the holiday, that way the rest of the group can back them up if something happens that will cause them to extend their trip and delay their involvement back into the project.

Contingency plan: Because the symptom is a physical problem, there is not much the team can do, except to do their best to prevent the risk from having too much impact to the project.

Insurance: The team leader would have to be ensuring that pair programing and code reviews are taking place.

Dropping the Class

Probability:

Minor

Impact:

Significant

Priority:

Minor

Risk: The team member dropping the class for reasons like they are in a very bad team environment, life issues, and many unforeseen issues.

Project impact: The impact to the project would depend on their position in the team, but either way it would be significant. Losing a member can sometimes destroy a project if it is not handled well.

Consequences: The major problem would be if the team member left the group without giving the team advance notice, the impact would be increase dramatically.

Response: The best ways to prevent dropping class from happening, and having too much impact is to try and find out each member's concerns and problems they might have with the project and in other fields, only if they are willing to share.

Contingency plan: Because the symptom is physical problem, there is not much the team can do, except to do their best to prevent the risk from happening and having too much impact to the project.

Insurance: The leaders would have to be trying to talk to the team member and keep updated on the status of the team members.

Hacking

Probability:

Minor

Impact:

Significant

Priority:

Minor

Risk: The most likely hacks are DDoS (Distributed Denial of Service), Trojan Horse (Disguised Software), Virus, Websites, and Worm.

Project impact: Slows down work flow, and delay due date, which messes with schedule, and add work load to the team. Potential loss of files.

Consequences: DDoS take away user's ability to use any network related task. Trojan Horse take away the user's ability completely and allow the hacker to do whatever they want with it. Virus try to attack the user's system. Malicious Websites help the other attack to get into your system. Worm is hidden attack that is a combination of DDoS and virus attack.

Response: Hacker hack your computer for a reason, so the way to avoid being hack is not to give that person the reason. Try not to make enemies, type your password in public, and have protection software active. Enemy may hack you out of spite, and just seeing the amount of digit the password have may give someone the curiosity to try.

Contingency plan: Immediate transfer of all your personal files, and project file to a removable storage. Then try to locate and remove the threat.

Insurance: Backup all the important files, and prepare removable storage.

Client Leaving

Probability:

Minor

Impact:

Severe

Priority:

Moderate

Risk: Client has a financial crisis, political disruption between the client and the team, or the client hires another team. The product lost value and client lost interest. Failure to meet a major part of the project.

Project impact: Cost the entire project, wasted time, progress, data, and money.

Consequences: Lose relationship, time, and money. Entire project dissolved.

Response: Come up way to compromise and allow the client to spend less. Show the client the product at different period to keep the client interested. Be as honest and direct as possible. Try not to fail on any major parts, and work hard to make up the damages if there were accidents. Keep in contact, don't neglect the client.

Contingency plan: Use the agreement to make a deal and come up with a plan that both party agrees with.

Insurance: Make a policy agreement that helps ensure the support of the client.

Due Date

Probability:

Moderate

Impact:

Significant

Priority:

Severe

Risk: Missing due dates. Approaching due dates with too many tasks. Underestimating the work load of the project and over estimating yourself. Failing to adjust according to the Triage Team's plan.

Project impact: May create a butterfly effect and make the subsequent due dates not be met as well, delaying other parts of the project. May increase the chance of other risks occurring. Causes stress to the team.

Consequences: Overdue hand in, decreases the team's morale, increasing stress and workload for the next part of the project.

Response: Create a check list to help identify what is the next item that needs to be finished. Create a calendar that notes all the important due date and events. Don't under estimate the work load, and manage your time accordingly.

Contingency plan: The Triage Team will take charge and the team will follow the emergency plan to accomplish as much as possible without decreasing the quality of the product for the upcoming due date.

Insurance: The Triage Team will create an emergency plan when the due date is approaching.

SIIM Innovation Challenge

Probability:

Moderate

Impact:

Significant

Priority:

Significant

Risk: Raising the standard of our program to present in public. It adds stress on the group, which can create an rushed program that has bugs.

Project impact: Adds a lot more work to the team, and extra things to do.

Consequences: Buggy program, forced commitment if we say something we don't have yet.

Response: Try to follow the schedule, and meet it to the best of our ability. Focus heavily on the visuals and functionality for the first part, to show off out program.

Contingency plan: If we didn't meet our goal of for the challenge, the stuff we did for the challenge can be translated to our normal project.

Insurance: Have a detailed schedule to follow, so we won't get off track in the midst of all the rush.

Busy Schedule

Probability:

Significant

Impact:

Significant

Priority:

Significant

Risk: Busy schedule caused by other classwork, which takes time away from spending on the project, affecting the estimated schedule, and the project's progression.

Project impact: Delaying parts of the project, can cause extra work for the other member of the group.

Consequences: Delaying the project schedule, which cause the other tasks of the project to be delayed as well.

Response: Try to take other classwork into account when estimating schedules, and try to tell someone or the project manager, if you're too busy and unable to complete something.

Contingency plan: Try to ask someone that is knowledgeable about the task to cover the work.

Insurance: Continue to peer program, and code review to spread the knowledge, so anyone is able to cover each other.

Time Estimation

Probability:

Severe

Impact:

Moderate

Priority:

Significant

Risk: Underestimation of times that will take to finish a task, which influences the task scheduling that is done by the project manager, in both good ways and bad ways depending on the estimation.

Project impact: Depending on the estimation, the effect can be huge or tiny. If the estimation is off by a couple of days, or a week, it would throw off the whole schedule.

Consequences: Delay the member that is waiting for the task to be finish and needs it soon. It can also create more work for the person, if the estimation had been large to begin with, the team would have move more members to the task.

Response: Take some time to do some research and think up a plan before estimating a time. Try not to under estimate too much, overestimating a little bit is better then underestimating.

Contingency plan: If the person doing their work finds out that it is going to take more work then they realised, then they should ask another team member if they can help.

Insurance: Just keep others informed on what is going on, so team member can give help when asked.

Important website cited:

Eye strain in VR:

https://essilorusa.com/content/essilor-usa/en/newsroom/news/virtual_reality_bad.html