Risk Documentation – Unit group

Homedork - Interactive Smart House

Name	Alphabetic representaion
Ali Habesh	A
Amr Al-Shaaba	B
Hani Al-zir	C
Stiv Abdullwahed	D

Revision History

Date	Version	Description	Author
2021-09-15	1.0	Modified Risk documentation.	A,B,C,D
2021-10-18	1.1	Re-checked on the risk list	A,B,C,D
2021-10-20	1.2	Fixing some grammar mistakes	A,B,C,D
2021-11-16	1.3	Removed a Risk from the Risk List and added server communication risk.	A,B,C,D
2021-12-6	2.0	No changes made, final version.	A, B, C, D

Risk List

Risk Description	Priority
R1. If the server is down, the user won't be able to use the application in order to communicate with his house.	High
R2. If not everyone gets their parts done in time for any reason it will affect the whole project negatively.	High
R3. Not coming to group meetings.	Medium
R4. Sickness of any kind.	Medium
R5. Unclear code may hurt the progress of the project.	Medium
R6. Gui bugs	Medium
R7. Comunication problem between the client and the server	High

Risk Handling Plans

R1. Prevention and management of risk 1

Impacts

If the server is down it will make the gui application non-responsive which would essentially make it worthless.

Indications

The website will be non-responsive, this will be the main indication once the problem occurs.

Mitigation Strategy

Focus all the resources on getting the server up and running, its possible to store some inputs in temporary files and once the server is up and running it will execute them. Also an emergency button should be added in case this happens (which is not part of the server).

R2. Prevention and management of risk 2

Impacts

If not everyone does what they are assigned the burden will increase on everyone since the others will have to help out to finish the assignments. This may cause irritation within the group. Can also halt the progress of the application.

Indications

The person never presents any finished code, or that they lie about how far they have come with the coding.

Mitigation Strategy

Make sure everyone knows exactly what they are supposed to do. Not present more tasks than the person can handle, and the most important part of all; ASK for help if you don’t seem to solve the problem, then the team will have to gather to try and solve the problem, since the project is about teamwork. The project leader will have to talk to the person about what to do about the problem.

R3. Prevention and management of risk 3

Impacts

If the person doesn't come to a group meeting, this will leave the rest of the group not knowing how far the person has come. It can also end up being dealbreaker in the end if the person has not achieved anything and the rest of the group must start doing his work from the start.

Indications

The person doesn't show up.

Mitigation Strategy

Make sure that every person at least leaves a textual update on how far they have come.

R4. Prevention and management of risk 4

Impacts

If the person is sick, this means he will not be able to show up which eventually means that this person's work needs someone else to take care of it. Can possibly slow down the progress.

Indications

The person is sick.

Mitigation Strategy

There is not much to do if someone is sick, but a good plan and dividing the work should be provided once someone has called in sick.

R5. Prevention and management of risk 5

Impacts

If code is not clean and well written with comments that clearly explains what each feature/method is supposed to do, it could affect the other tasks if they require that specific code to work. Badly written code can also create problems for the actual running time making the application less responsive.

Indications

Code that doesn't clearly show what goal it is trying to achieve and its hard for someone else to continue working on it

Mitigation Strategy

Write clear code commenting the important lines and short description next to method names on what the developer is trying to achieve. Other solutions such as group meetings can also come in handy.

R6. Prevention and management of risk 6

Impacts

will make the application feel unfinished which will make the user unsatisfied of the product.

Indications

Gui is not working smoothly.

Mitigation Strategy

Always try the gui out when adding a new future to it, testing is necessary.

R7. Prevention and management of risk 7

Impacts

This may delay the team from their original deadlines depending on how hard it is to find and fix the problem that is preventing the communication, may also hurt the morale and stress the developers. If a problem arises while the user is using the application it will give an unsatisfactory experience.

Indications

The client can not comunicate with the server.

Mitigation Strategy

Testing the communication with a dummy server to try to minimize the chance of problems when actually testing with the server.