Revision History

Interactive House

Name	Associated Letter
Lukas Olsson	A
Wills Ekanem	B
Bujar Rabushaj	C
Besnik Rabushaj	D

Date	Version	Description	Author
16/9/2021	1.0	Talked about the potential risks of the project.	A, B, C, D
6/10/2021	1.1	Secondary Revision	A, B, C, D
23/10/2021	1.2	Third revision. Added more risks. R4, R7, R8.	C, D
12/11/2021	1.3	Fourth Revision. Added Changes in risk priorities subchapter and altered risk levels according to current state	C, D

Risk List

Risk Description	Risk Level
R1. Distance Coding	Medium
R2. Communication between groups	Medium
R3. Covid Risk	Low
R4. Code complexity	Very low
R5. Bug Testing	Very low
R6. Version Management System	Very low
R7. Internal planning	Very low
R8. A team member dropping out	Very low

R1. Prevention and management of risk 1

Impacts

Distance coding could make it more difficult to stay on the same page when working on the project.

Indications

We aren't meeting regularly and have difficulty understanding what others are working on.

Mitigation Strategy

Regular meetings online to update each other on what we are working on as well as a text group where we can stay up to date or ask questions.

R2. Prevention and management of risk 2

Impacts

With the groups not always being in full contact with each other it could be difficult to stay up to date with what they are planning or working on. If they decide to change the scope on any aspect of their work and don't tell us it could leave the project in jeopardy.

Indications

Not knowing how the other groups are working/doing.

Mitigation Strategy

Regular meetings online to update each other on what we are working on as well and if we need to have a more conventional meeting to update each other.

R3. Prevention and management of risk 3

Impacts

If any of the group members either get sick or we can't have meetings in person due to risks from covid it could affect the projects outcome.

Indications

If anyone becomes sick with covid or is unable to show up to meetings if they think they may be sick.

Mitigation Strategy

Have more online than in person meetings to stay safe while we stay up to date.

R4. Prevention and management of risk 4

Impacts

Since we are operating in groups it is important that everyone can understand the code so that various people can take over or continue with that part if need be. Because if the code is too complex it may be impossible for the person taking over to understand the code.

Indications

A good indication could be that nothing gets done in time.

Mitigation strategy

The group can discuss beforehand at what level each person is when it comes to coding, and they can then adjust the future code based on that information.

R5. Prevention and management of risk 5

Impacts

Without access to the other groups code or the physical devices it may be more difficult to bug test specifically to them.

Indications

If we are having trouble with the testing or if it all works prior to testing the actual real devices or other groups code.

Mitigation Strategy

Keep up to date on how the other groups are working on their parts as well as understanding what information will be required for the server to function.

R6. Prevention and management of risk 6

Impacts

If any of us have difficulty with GitHub, then it could affect how much an individual contributes to the project and therefore the project itself could suffer.

Indications

If someone hasn't updated their branch in a while or they don't merge regularly.

Mitigation Strategy

Keep your branch up to date and merge regularly and do not be afraid to ask for help with GitHub.

R7. Prevention and management of risk 7

Impacts
If we do not plan what we must accomplish in the upcoming weeks properly, it might lead to complications such as not being able to finish certain features.

Indications
If the subgroup is approaching a deadline without much work completed.

Mitigation strategy
Making sure beforehand that every step is thoroughly planned so the group can account for time.

R8. Prevention and management of risk 8

Impacts
It would increase the workload by a lot for one person enormously since our subgroup has divided the coding into client/server respectively.

Indications
We would be notified if that was the case.

Mitigation strategy
The other team members would make sure to cover for the one that has the increased workload.

Changes in priorities of risks
Since this project is approaching its final stage, many of the risk levels have naturally changed. Basically, most of the risk levels have been reduced one or more levels since most of the project is almost fully functional and nearing completion. Alternatively, if the project had not proceeded well, there might have been a need to put a lot of the risks higher. For example, if the code was not being completed within the respective deadlines, maybe the internal planning risk should be set to high instead of low because the resources may have not been allocated correctly from the start. So, the risks and their respective risk levels can alter significantly throughout the project's runtime depending on the specific project.