Category	Description	Likelihood (1-9)	Impact (1-9)	Importance (C*D)	Preventive actions	Remedial actions	comments
communication	Miscommunication with the customer	6	5	30	Confirm any and all changes or decissions about the project explicitly with the customer	Costly and time consuming changes to the project	
communication	Conflicts within the group	3	5	15	Beer	More beer	
design	Design is too complicated	5	8	40	Redesign with simpler ideas and less functionality.	drop functionality and make things work.	
design	Design is too simplistic	3	4	12	Specify more.	add functionality	
exterior	Illness	5	4	20	Eat healthy, sleep enough, be well clothed.	Stay home and sleep a lot to get better.	Probably not going to become a problem but, you never know.
exterior	Disruptive facilities	5	2	10	Book isolated rooms	take a break and find a new location to work	
exterior	End of the world	0.000001	10	0.00001	Sacrifice goats	None, it won't happen before december, and we'll be done in may	to be removed later. ("The hell it is!" ~Stig Tore)
group	Inability to work under pressure	4	5	20	Know who and make sure they don't get stressed out.	calm people down and take a break.	
group	Team unfamiliar with the type of project	7	2	14	Time and research	Put more hours in to the project to familiarise ourselves more with the project type.	
group	Inefficient team structure	1	3	3	Evaluate pros and cons for the given structure. And see if it is necessary to create a more elaborate structure for the group.	Restructure the team and assign roles and responsibility areas	
organization	Lack of Room. No work space for us.	6	5	30	Book rooms in time and possibly work from home.	Find an alternative room on showing up at Gløs. The backup plan is to be at Drivhuset in the red room.	This has been a bit of a problem as you can't book a room for five weeks at a time Monday through Thursday between 1000 and 1600.
planning	Cascading delays	5	9	45	Stress mastering and beer to calm the nerves.	Efficient time planning. Ultimately drop functionality to complete necessities.	
planning	Optimistic scheduling	5	8	40	Regularly evaluate the schedules and make changes as necessary	**Overtime for everybody!!!**	
planning	Paperwork overhead is too big	4	9	36	Use more time on paperwork throughout the project.	Cut down on the paperwork and focus on the product.	
planning	Project too large in required effort or code size	6	6	36	Minimize the project and cut off features and components that is not needed.	Minimize the project and cut off features and components that is not needed.	
planning	Too coarse-grained requirements	6	5	30	Work with the customer to specify requirements better	Communicate with the customer to get a new set of must have requirements.	
planning	Incomplete schedule	4	7	28	Regularly evaluate the schedules and make changes if necessary	Add elements to the schedule and insert extra empty timeslots for unforeseen work.	
planning	Additional requirements turn up	7	4	28	Make sure all requirements are found before starting	Overrule the requirements that came later as optional.	
planning	Schedule slips without being discovered	3	7	21	Find up to date information about the course, project and deadlines and share them in the group, so all the members will remember.	Reschedule and make up for lost time.	
planning	Faulty planning	3	6	18	Make sure we properly research things before we decide anything	Improvise or bang head into the wall	
planning	Miss deadlines	2	4	8	Follow the plan and make sure we are ahead. Plan buffers.	Use planned buffer time to cath up. Use the weekend if necessary.	
tech	Integration with external libraries more complicated than expected	7	7	49	Prestudy	Re schedule and use more time then expected. Buffer zones are to be used if this happens.	
tech	Reimplementation due to faulty design	5	6	30	Use time on the design process and make sure the design is right.	Skip functionality and work overtime.	
tech	Poor code-quality in external libraries	5	6	30	Try to only use well-tested libraries, stay away from the highly experimental ones. Write code ourselves.	Find an alternative or create it your self.	
tech	Failure to implement chosen technologies	3	9	27	Proper research into the technologies we are using and a proper understanding of those	Acquire knowledge and ask questions	This could be very bad if it should occur, but with proper research and a good understanding it should not be a problem

tech	External libraries not suited for project	5	5	25	Research library content and usage.	Find an alternative library or implement the necessary code.	
tech	Unfamiliarity with the core technology of the design	8	3	24	Research and training.	Research and training. Ask experts for help.	
tech	Reliance on unfinished software	3	8	24	Be smart, unstable releases is catgorized as "Do not use!"	Make sure the relied on software is tested and implemented first.	
tech	Unfamiliar software or hardware environments	5	3	15	Time and research	Use the buffer zones.	
tech	Hardware failure	2	5	10	Maintain and take care of equipment, keep backups of important stuff	Acquire and set up new, or fix old, equipment as soon as possible	proper use of Git should minimize loss of work
tech	Broken codebase	1	8	8	Use Git (distributed as opposed to sentralised SVN). Keep one or more testing branches, which are merged with the Master branch only after having passed a full and rigourous test suite	Don't panic	Probably not going to occur, as we use Git and will at all times do development against a testing branch
tech	Can't get Identity Server to work	3	7	21	Do proper research around our alternatives and comprehensive evaluation of WSO2's Identity Server	Talk to the customer to come to some agreement	