Risk Table Initial Assessment

ID	Risk name	Type of risk	Description	Probability	Impact	Risk factor	Mitigation	Responsible	Follow-up date	Status date	Status
R1	Incorrect estimation of tasks	Project risk	Overestimation or underestimation, exclusion of team members from estimation process	3	5	15	Using Jira with Scrum for establishing Sprints, a good testing schedule has to be created inside Jira.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R2	Team members are unfamiliar with the technologies chosen	Project risk	The developers are inexperienced they might not be familiar with the chosen tech stack or they need more time to get familiar with it.	2	4	8	Tech stack choice that all developers are atleast comfortable in code writing in.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R3	Unable to cooperate as a team	Project risk	Team members may not have worked together a lot, and diffrence in personality and project view can hurt the project.	3	3	9	A third person or arbiter should be brought when conflicts happen. Setting clear goals for what the project should be and at what point are crucial	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R4	Illness	Project risk	Illness of the team members can slow down the project a lot as the number of team members is small.	3	4	12	Team members should be exercising a decent amount and protecting their health from catching colds or other illnesses.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R5	Wrong definition of requirements	Project risk	The team wasn't able to gather requirements well enough from the customer and the user stories weren't descriptive enough, missing key features for the customer or requirements that customer didn't want as much or at all were prioritized.	2	5	10	A formal review of the requirements will help with establishing key features, how those features look and what has to be done to achieve them. Speaking again with customer can also be required.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R6	Bad documentation	Product risk	The team doesn't document well enough their code causing errors in function usage and regressions	1	4	4	Using a code standartizer(either a document stating standarts or a module which helps with this) can fix this, code reviews should always check for documentation.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R7	Website has a lot of time to load	Product risk	The system design and architechture weren't design good, causing delays in the system usage	2	4	8	Stress performance testing will help with this issue identify this issue, choosing the correct technologies can also be of benefit	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R8	Security vulnerabilities not addressed	Product risk	No protections added for SQL injection, server side request forgery or cross-site scripting causing massive damage to system and product	3	5	15	Code reviews can check if security has been addressed. Integration tests and system integration tests can check if our security has been done correctly	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R9	Lack of user validation	Product risk	End product doesn't get presented to users for validating their opinion on the end product	4	5	20	Doing a gorilla test can help as it is quick to do, giving adecuate time in the sprints for an acceptance test by the users of implemented features.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R10	Errors in calculations	Product risk	Misunderstanding in requirements causes errors or misunderstanding in how calculations have to be done for some use cases	2	4	8	A formal review of the requirements will help with establishing key features, how those features look and what has to be done to achieve them. Speaking again with customer can also be required.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R11	Key feature not implemented	Product risk	Failure to addhere to testing and coding schedule and lack of knowledge of team members are the causes to failing to implement the feature on time.	3	5	15	Good tech stack choice, well thought-out sprint schedule with enough time for all types of testing. Good prioritization of tasks has to be done.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R12	Leaving a defect in production	Product risk	Unit tests and integration tests design were not done correctly missing or misunderstanding what had to be done.	1	5	5	Formal review of requirements, code reviews at the end of sprints to check. Acceptance testing before sprint end can help identify issues before release into production.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk
R13	Missing deadline	Project risk	Team members weren't able to keep up with the schedule.	2	5	10	Scrum schedule created. Change schedule if it looks like key features are being delayed and deadline is closing in. Overtime work can also help.	Aleksandar Miroslavov Minchev	02/06/2022	30/05/2022	Initial assessment of risk

	Impact 1: Not significant 2: Minor 3: Moderate 4: Major 5: Severe	Probability 1: Rare 2: Unlikely 3: Possible 4: Likely 5: Almost Certain				
	Probability	Risks				
	Almost Certain					
	Likely					R9
	Possible			R3	R4	R1,R8,R11
	Unlikely				R2,R7,R10	R5,,R13
	Rare				R6	R12
	Impact	Not significant	Minor	Moderate	Major	Severe

Risk Table Mid project assessment

ID	Risk name	Type of risk	Description	Probability	Impact	Risk factor	Mitigation	Responsible	Follow-up date	Status date	Status
R1	Incorrect estimation of tasks	Project risk	Overestimation or underestimation, exclusion of team members from estimation process	2	5	10	Using Jira with Scrum for establishing Sprints, a good testing schedule has to be created inside Jira.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	Jira dashboard created and task were defined and assigned to each team member. Equal split team members, with considerable diffrence in difficulty
R2	Team members are unfamiliar with the technologies chosen	Project risk	The developers are inexperienced they might not be familiar with the chosen tech stack or they need more time to get familiar with it.	1	4	4	Tech stack choice that all developers are atleast comfortable in code writing in.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	All team members use/know a decent amount of Javascript, so Javascript based tech chosen (NodeJs, React,MySQL)
R3	Unable to cooperate as a team	Project risk	Team members may not have worked together a lot, and diffrence in personality and project view can hurt the project.	2	3	6	A third person or arbiter should be brought when conflicts happen. Setting clear goals for what the project should be and at what point are crucial	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	Jira dashboard helped with scheduling and setting clear goals, daily scrum meetings also assisted in managing conflicts. No arbitration was required so far.
R4	Illness	Project risk	Illness of the team members can slow down the project a lot as the number of team members is small.	2	4	8	Team members should be exercising a decent amount and protecting their health from catching colds or other illnesses.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	Team members keep a healthy schedule.
R5	Wrong definition of requirements	Project risk	The team wasn't able to gather requirements well enough from the customer and the user stories weren't descriptive enough, missing key features for the customer or requirements that customer didn't want as much or at all were prioritized.	1	5	5	A formal review of the requirements will help with establishing key features, how those features look and what has to be done to achieve them. Speaking again with customer can also be required.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	Requirements specification document was created and an analysis of it was conducted, ironning out any questions from delevopers.
R6	Bad documentation	Product risk	The team doesn't document well enough their code causing errors in function usage and regressions	1	4	4	Using a code standartizer(either a document stating standarts or a module which helps with this) can fix this, code reviews should always check for documentation.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	During Daily Meetings documentation check for completed functions was done. ESlint was implemented to guarantee coding standarts.
R7	Website has a lot of time to load	Product risk	The system design and architechture weren't design good, causing delays in the system usage	2	4	8	Stress performance testing will help with this issue identify this issue, choosing the correct technologies can also be of benefit	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	JMeter choosen to do Stress Performance. Created a tests plan for stress performance testing.
R8	Security vulnerabilities not addressed	Product risk	No protections added for SQL injection, server side request forgery or cross-site scripting causing massive damage to system and product	2	5	10	Code reviews can check if security has been addressed. Integration tests and system integration tests can check if our security has been done correctly	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	During Code reviews checks for security were done. Integration testing helped identify key issues.
R9	Lack of user validation	Product risk	End product doesn't get presented to users for validating their opinion on the end product	3	5	15	Doing a gorilla test can help as it is quick to do, giving adecuate time in the sprints for an acceptance test by the users of implemented features.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	No actions taken for gorrila testing. Given to time to do acceptance test before starting next spring.
R10	Errors in calculations	Product risk	Misunderstanding in requirements causes errors or misunderstanding in how calculations have to be done for some use cases	1	4	4	A formal review of the requirements will help with establishing key features, how those features look and what has to be done to achieve them. Speaking again with customer can also be required.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	Requirements specification document was created and an analysis of it was conducted, ironning out any questions from delevopers. Questions towards busines calculations answered.
R11	Key feature not implemented	Product risk	Failure to addhere to testing and coding schedule and lack of knowledge of team members are the causes to failing to implement the feature on time.	2	5	10	Good tech stack choice, well thought-out sprint schedule with enough time for all types of testing. Good prioritization of tasks has to be done.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	Schedule was created and is adhered to. Tech stack that all team members know was choosen.
R12	Leaving a defect in production	Product risk	Unit tests and integration tests design were not done correctly missing or misunderstanding what had to be done.	1	5	5	Formal review of requirements, code reviews at the end of sprints to check. Acceptance testing before sprint end can help identify issues before release into production.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	Review of requirements conducted, time given for acceptance testing in scrum schedule, code reviews were done at the end of sprints
R13	Missing deadline	Project risk	Team members weren't able to keep up with the schedule.	2	5	10	Scrum schedule created. Change schedule if it looks like key features are being delayed and deadline is closing in. Overtime work can also help.	Aleksandar Miroslavov Minchev	05/06/2022	02/06/2022	Scrum schedule created. No changes to it have been required so far. No overtime work as well.

Impact	Probability
1: Not significant	1: Rare
2: Minor	2: Unlikely
3: Moderate	3: Possible
4: Major	4: Likely
5: Severe	5: Almost Certain

Probability	Risks				
Almost Certain					
Likely					
Possible					R9,
Unlikely			R3	R4,R7	R1,R8,R11,R13
Rare				R2,R6,R10	R5,R12
Impact	Not significant	Minor	Moderate	Major	Severe

Risk Table Final Assessment

ID	Risk name	Type of risk	Description	Probability	Impact	Risk factor	Mitigation	Responsible	Follow-up date	Status date	Status
R1	Incorrect estimation of tasks	Project risk	Overestimation or underestimation, exclusion of team members from estimation process	1	5	5	Using Jira with Scrum for establishing Sprints, a good testing schedule has to be created inside Jira.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R2	Team members are unfamiliar with the technologies chosen	Project risk	The developers are inexperienced they might not be familiar with the chosen tech stack or they need more time to get familiar with it.	1	4	4	Tech stack choice that all developers are atleast comfortable in code writing in.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further were taken
R3	Unable to cooperate as a team	Project risk	Team members may not have worked together a lot, and diffrence in personality and project view can hurt the project.	1	3	3	A third person or arbiter should be brought when conflicts happen. Setting clear goals for what the project should be and at what point are crucial	Aleksandar Miroslavov Minchev	05/06/2022	None	No further were taken
R4	Illness	Project risk	Illness of the team members can slow down the project a lot as the number of team members is small.	1	4	4	Team members should be exercising a decent amount and protecting their health from catching colds or other illnesses.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further were taken
R5	Wrong definition of requirements	Project risk	The team wasn't able to gather requirements well enough from the customer and the user stories weren't descriptive enough, missing key features for the customer or requirements that customer didn't want as much or at all were prioritized.	1	5	5	A formal review of the requirements will help with establishing key features, how those features look and what has to be done to achieve them. Speaking again with customer can also be required.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R6	Bad documentation	Product risk	The team doesn't document well enough their code causing errors in function usage and regressions	1	4	4	Using a code standartizer(either a document stating standarts or a module which helps with this) can fix this, code reviews should always check for documentation.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R7	Website has a lot of time to load	Product risk	The system design and architechture weren't design good, causing delays in the system usage	1	4	4	Stress performance testing will help with this issue identify this issue, choosing the correct technologies can also be of benefit	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R8	Security vulnerabilities not addressed	Product risk	No protections added for SQL injection, server side request forgery or cross-site scripting causing massive damage to system and product	1	5	5	Code reviews can check if security has been addressed. Integration tests and system integration tests can check if our security has been done correctly	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R9	Lack of user validation	Product risk	End product doesn't get presented to users for validating their opinion on the end product	1	5	5	Doing a gorilla test can help as it is quick to do, giving adecuate time in the sprints for an acceptance test by the users of implemented features.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R10	Errors in calculations	Product risk	Misunderstanding in requirements causes errors or misunderstanding in how calculations have to be done for some use cases	1	4	4	A formal review of the requirements will help with establishing key features, how those features look and what has to be done to achieve them. Speaking again with customer can also be required.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R11	Key feature not implemented	Product risk	Failure to addhere to testing and coding schedule and lack of knowledge of team members are the causes to failing to implement the feature on time.	1	5	5	Good tech stack choice, well thought-out sprint schedule with enough time for all types of testing. Good prioritization of tasks has to be done.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R12	Leaving a defect in production	Product risk	Unit tests and integration tests design were not done correctly missing or misunderstanding what had to be done.	1	5	5	Formal review of requirements, code reviews at the end of sprints to check. Acceptance testing before sprint end can help identify issues before release into production.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken
R13	Missing deadline	Project risk	Team members weren't able to keep up with the schedule.	1	5	5	Scrum schedule created. Change schedule if it looks like key features are being delayed and deadline is closing in. Overtime work can also help.	Aleksandar Miroslavov Minchev	05/06/2022	None	No further actions were taken

	Impact 1: Not significant 2: Minor 3: Moderate 4: Major 5: Severe	Probability 1: Rare 2: Unlikely 3: Possible 4: Likely 5: Almost Certain				
	Probability	Risks				
	Almost Certain					
	Likely					
	Possible					
	Unlikely					
	Rare			R3	R4,R6,R2,R7, R10	R12,R9,R1,R8, R11,R5
	Impact	Not significant	Minor	Moderate	Major	Severe