

SHOWROOM

Risks Management

(by Diego Movilla)

Risky Python: Luis Jesús Cepeda

Jaime Antolín

Álvaro López

Francisco Gómez

Diego Movilla

Index:

1. Team 'Risky Python' Risks Management	2
a. Personnel shortfalls	2
b. Unrealistic schedules and budgets	2
c. Developing the wrong software functions	3
d. Change of teacher	3
e. Technological risks	4
f. Carry on an inconsistent project	4
g. Lack of commitment	5
h. Evaluation of project cost	5
i. Client's modifications	6
2. Follow-up of the previous document	7
3. Team 'Tgivial' Risks Management	8

Team 'Risky Python' Risks Management:

"This pages include a list of risks, following some of the "Typical S.E risks (Boehm 1989)" which may or not may happen but must always bear in mind when developing the project. Starting with a short follow-up of the evolution report with respect to previous treatments of risks (new risks arising, risks overcome, changes in risk assessment, contingency plans applied successfully or not...). At the end, there are a few arising risks we have found when developing our tasks." Risks are categorized by probability and impact.

--Starting with a re-analyzed of some previous itineration risks (1 to 5). Continuing with new arising risks (6 to 9).

1. Personnel shortfalls:

The personnel of an organization are the people who work for it. Having problems with employees can mean a lack of staff for doing a certain project, staff not qualified enough for doing certain issues, lack of communication between members etc.

- Probability:

Really High

- Impact:

Catastrophic

- Contingency plan:

As we are a small, beginner group, dealing with a big project may seem pretty difficult. Therefore, we can make a smaller version of our project which can also satisfy and fulfill what we are aiming to. In order to do so, and to avoid wasting useful time, among other things, we will design a communication plan where we will establish every member's role during the project development.

2. Unrealistic schedules and budgets:

Risk which means an underestimation of some of the functionalities of the project, an underestimation of the time that we have. We could also face difficulties with time because of mobility of resources, insufficient resources, changes in pre-set dates and changes in requirements.

- Probability:

From Really high to Medium

- Impact:

High

- Contingency plan:

(Well applied until this point. We have applied a reduced version and have organised such as we encompass all our tasks in a realistic time period. We have focused on main parts so we fulfill the requirements of the project. In advance, if we have extra time, we could think about including some improvements.)

As written in risk 1, the main plan to avoid schedule and communication problems is to organize before starting the role every member should take and elaborating a timeline where we set deadlines. We should also take into account possible delays so we can have enough time to fix and continue without serious problems. Planning and monitoring the update project is a must-do.

3. **Developing the wrong software functions:**

Developing wrong software functions may occur since there is not a clear view of the project and therefore no visibility of its evolution.

- Probability:

Medium

- Impact:

Really high

- Contingency plan:

In order that not only the main members of the group but all of them have a clear view of what we are aiming to, it is essential to establish periodic follow-up meetings to report on the progress of the project.

4. **Change of teacher:**

In the second half of the course we will have a new teacher who may ask for new labours.

- Probability:

Medium

- Impact:

Low

- Contingency plan:

Even though the teacher might change, it is unlikely that he asks for a completely different thing which obliges us to change many things in our project.

5. **Technological risks:**

Using software or other inadequate computer utilities will mean a decrease in our productivity. If there are technological problems, this will delay the delivery of the projects.

- Probability:

Low

- Impact:

Medium

- Contingency plan:

(Last contingency plan applied successfully. We updated our project to new programs which helped us to develop some tasks, such as 'Modelio'.)

Nowadays, with the possibility of searching in internet to find the best softwares for our respective tasks we have to do, finding problems due to a misuse of a program or due to a bad choice could be pretty stupid. Therefore, before starting, we should get informed about the best and most efficient programs and softwares to run our projects.

6. **Carry on an inconsistent project:**

Delegating work to other members or groups is really common when developing big projects. Therefore, it is crucial to -even though you know you will just work on something for a certain time- perform every task in the best way as possible, as you would like to find it. Working in a big team means you will work on things by your own but always having in mind that you are part of that team and that whatever affects the project may affect you in the future.

- Probability:

Really high

- Impact:

Catastrophich

- Contingency plan:

Personally, when developing Showroom's activity, use case and sequence diagrams, we found a really poor specification, bad relation between use cases and incomplete CRC Cards. Therefore, we could not make our corresponding task and had to redo everything from the beginning in order to be able to do our work.

7. **Lack of commitment:**

We may find that the client we are working for does not contribute (payments, finances etc...) as the team expected.

- Probability:

Medium

- Impact:

High

- Contingency plan:

As mentioned in almost every risk, it is essential to establish before starting, a global view of the task that each group will perform and, of course, the role that the client will play so there is no confusion in the future.

8. **Evaluation of project cost:**

Even though it may seem obvious, if we get to a point where in order to move forward we need a budget which we realise we do not have and therefore we get stuck because of a lack of planning.

- Probability:

Medium

- Impact:

High

- Contingency plan:

Even though the cost of a project may change during its development, we should estimate the overall cost of the whole project and each of its parts.

Having emergency plans for situations like this is also really advisable if we do not want to lose time in the future. In order to guarantee not to waste useful money, it is important to eliminate those tasks that contribute less to the whole project and try to get a result that meets the minimum requirements necessary to consider that the project has been successful.

9. **Client's modifications:**

Throughout the development of a project, it can undergo some changes. The project can grow in complexity, i.e. the client adds new requirements. However, this type of changes are common nowadays as projects must adapt to the needs of the market which is in constant change.

- Probability:

Medium

- Impact:

High

- Contingency plan:

It is important to keep track of the market. To do so, it is useful to have a member in the group who can predict future changes that may affect the project to prevent these modifications from taking us by surprise.

Follow-up of the previous document:

- (Risks 1 & 3) -> Personnel shortfalls & Unrealistic schedules:

3 months later, we have adjusted the initial version and story of our project to the time we have had to develop it. As the contingency plan said, a reduced, more realistic and a simpler version has been carried out. Therefore, schedules' risks have been mitigated.

- (Risk 6) -> Management change:

3 months ago, the probability of changing the teacher was not important. However, we had the information that he/she might change some project development criteria. Therefore, the risk increased in priority and in rank due to its increase in probability.

- (Risk 8) -> Technology change:

When the last iteration was made, we did not know about the existence of programs like "Modelio". Therefore, we did not know how would we develop graphically some tasks. However, when this was presented to us, we learned and adapted our project to it.

Nowadays, other risks may play a more important role than when this document was written and most risks might remain the same (specified at the beginning), basically because they encompass long-term risks, which can not be analyzed yet.

Team ‘Tgivial’ Risks Management:

Those are the main risks that our team has identified so far. This first page contains the most important risk to take into account. All of them were identified in group.

1- Personnel shortfalls :

Probability: **High**

Priority: **Extremely high**

Type: People/Estimation

(Affects Project and product)

Plan:

We believe that this is a very big project and we are only four members. Since our classmates underestimated the size of their idea, we should find a way to satisfy them even though we could not finish it.

In order to do that we are willing to elaborate a reduced version of the project with a more realistic vision that still pleases the client.

As we are going to take out some functionalities that are not essential, in the case that everything works perfectly and we have more time (which is unlikely) we would add those extra parts.

2- Developing the wrong software functions: Related with a size underestimation.

Probability: **High**

Priority: **High**

Type: Technology/Requirements

Plan:

There are many functionalities drawn in the interface that are not correctly explained.

Examples: Trending, News, Upcoming Shows... Should we create does part or link them to elsewhere (?).

In order to fix this we will follow the following protocol: If it is not included in the description, we will link them to other webpages so we do not have to develop them.

Plus, who are we supposed to change the language in the web-page? Using Google Translate? Writing everything twice? How many languages?

There is no fixing-part interface defined either a forum graphical description!

In UC-4 there is no further explanation about how to add, delete, etc something.

We have no “guideline” for the forum at all. We should have one.

3- Unrealistic schedules and budgets:

Probability: High

Priority: High

Type: Estimation

Plan: This is directly related to risk 1 and the way to mitigate this risk is mostly the same.

4- Developing the wrong user interface:

Probability: Medium

Priority: Medium

Type: Requirements

Plan: If we are not sure about some font/color we will directly ask the client.

5- Gold plating:

Probability: Low

Priority: Insignificant

Type: Organisational

6- Management change:

Probability: We will change our teacher for sure by second semester

Priority: Insignificant

Type: Management

Plan: We will assume that this new teacher will not change grading-criteria.

7- Case tool underperformance:

Probability: Occasional

Priority: Low

Type: Technologies

(Affects the products)

8- Technology change:

Probability: Improbable

Priority: Insignificant

Type: Technology

(Affects the business)

Plan:

We think that the technologies can not change so much in such a short time.

In case that the technologies change the project must be adapted to that new technologies.

But in series or cinema apps there is not much to improve, the things that could be improved are things like the resolution, the interfaces, ...

9- Product competition:

Probability: Improbable

Priority: Insignificant

Type: Technology

(Affects the business)

We include this valuations as well so the priority of the risk we have included makes more sense:

Basic elements		
Story for the application with actors and use case identifiers	8	(Database must not be included in the list of actors)
List of actors	8	(Database must not be included in the list of actors)
List of use cases	8	(UC4 does not specify the action that has to be done)
Relation between use cases	9	
Detailed sequences (main and alternative) for each use case	9	
Non functional requirements for each use case	9	
Glossary	10	
User interface prototypes for each use	7	(Forum is not specified)(Settings is not specified)
Goal satisfaction		
Does it provide a general vision of the system?	6	(There are several things that it is not specified how they want it)
Does it allow details to be annotated wherever needed?	5	(There is not enough space to write annotations)
Does it help us to decide what we are going to do when?	8	
Is it easy to convert into code?	7	
Objective data on the project		
Is it clear who has worked on this up to this point?	10	
Is it clear how much work has been done by each participant?	10	
Is it clear how much work has each part of the specification taken?	0	(It is not specified the time at all)