

Always know what can go wrong

The Table XI software risk assessment template

Risk is a part of every software development project, no matter how much we may want to ignore it. So instead of averting our eyes and hoping for the best, we came up with a way to address risk head-on. By putting numbers around both the qualitative risks we can't see and the quantitative ones we can, we can go into projects knowing where the quicksand is, and direct the right resources at plugging those holes.

1 Define the project.

Before you can quantify the risk, you need to break the project down into epics and stories. These will tell you exactly what functions the software needs to perform. Assigning points that show the relative complexity of each story lets us get a sense of the project scale. (More on story points in Step Three).

That's not enough to show risk though — we need to assign meaning. To do that, we rank each story as high, medium or low priority. If a story is risky, but low priority, we may put it off, knowing it doesn't have to be completed. If it's necessary and high risk, then we know where to direct our efforts.

2 Visualize the qualitative risk.

To understand risk from the client's perspective, we take the above list and have client stakeholders give each story a ranking of high, medium or low based on three factors: completeness, volatility and complexity. That tells us if the story is fully fleshed out, if it's likely to change based on business decisions and how relatively complicated it is. That gives us risk.

Qualitative risk analysis

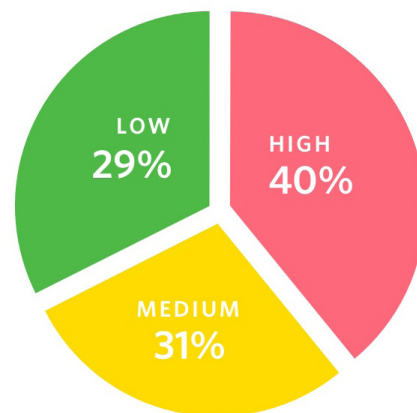
How story completeness, complexity, and volatility estimates risk

	Relative completeness		Relative volatility		Relative complexity		Estimated risk
Matching	UNKNOWN	+	LOW	+	COMPLEX	=	HIGH
Referral	INCOMPLETE	+	HIGH	+	COMPLEX	=	HIGH
Candidate profile	INCOMPLETE	+	LOW	+	COMPLEX	=	MED
Account	COMPLETE	+	LOW	+	STANDARD	=	LOW
General	COMPLETE	+	LOW	+	SIMPLE	=	LOW

From there, we need to understand what that means for the project as a whole, and what that means for each collection of stories that makes up a functional area (e.g., the 10 stories that define the functionality needed for a sign-in process). We get the first by averaging the risk of each story and totalling them together to get an overall risk profile of the project, seen here.

We get the second by averaging the risk of every story in a functional area, then plotting them on a graph. The size of the bubble is determined by the total story points in the functional area, letting us know how complex it is.

Overall estimates risk



Estimate risk analysis

Where should your product team dig deeper?



That top right quadrant? That's where the dragons are. We start our projects there.

[Read more about how we assess risk on projects](#)

3

Measure the quantitative risk.

While we're getting all the client's cards on the table and visualized, we kick the project to our developers so they can do the same. This is how the story points are determined. Developers and Designers work together to estimate a minimum, likely and maximum number of story points. To quantify it accurately — remember, one of these story points is more likely than the other two — we use the following equation:

$$\frac{(1 \times \text{minimum}) + (4 \times \text{likely}) + (1 \times \text{maximum})}{6}$$

For some stories though, the Maximum number of story points is more likely than the Minimum, because our clients have flagged those stories as especially risky in the qualitative analysis. For those stories, we use this equation:

$$\frac{(1 \times \text{minimum}) + (4 \times \text{likely}) + (3 \times \text{maximum})}{8}$$

That weighted average is what we ultimately assign in the way of story points. We also record the delta between the Minimum and Maximum estimates, so we know what level of risk the developers see for each story, as you can see in the Master Story List below.

STORY NUMBER	STORY TITLE	MINIMUM ESTIMATE	LIKELY ESTIMATE	MAXIMUM ESTIMATE	ESTIMATE VARIANCE
45	Match candidates and job opportunity	5	8	13	7
46	Display job opportunity matches	1	2	3	2
47	Notify candidates of matches	2	3	5	3
48	Notify employer of matches	2	3	5	3
49	View match details	1	2	2	1

To take action on that information, we bundle it up the same way we do the qualitative information, collecting stories into functional areas and averaging the risk — in this case the estimate variance.

Estimate variance by functional area



4 De-risking the project.

With the quantitative risk measured on the estimate variance chart and the qualitative risk quadrant, we have a full picture of the risk on a project. When we know how likely we are to encounter quicksand — and where — we're able to better staff the project and strategically tackle functional areas to de-risk the project. As we reduce the risk, we also reduce the need for contingency, giving us time and money to spend on additional features, or to shave off the project altogether. Most importantly, modeling risk at the start of a project creates an honest, open understanding with the client about what needs to be done.

[See how we use these risk charts to run projects](#)