Validación de Proyectos de Sobrevivencia
Administración de Proyectos de Sistemas de Información

Instructions

Fill in the "Project Name" at the top of the sheet

For each question, give the project 3 points for each "yes" answer, 0 points for each "no" answer.

Give the project partial credit if you feel that is most accurate—for example, give it 2 points for "probably" and 1 point for "kind of, but not really."

If the project is in the early stages, answer the questions based on the project plans.

If the project is underway, answer the questions based on what is actually happening on the project.

The section following the test explains how to interpret the score.

Incomplete answers are colored blue

Answers outside the valid input range are colored red.

Valid, complete answers are colored black

The "Tools | Protection | Unprotect Sheet" command will unprotect the entire worksheet, allowing you to customize it for your projects

Click the "Survival Worksheet" tab, below, to begin.

Validación de Proyectos de Sobrevivencia
Administración de Proyectos de Sistemas de Información

Survival Test

Project Name:	StartUpBase – Plataforma de Incubação Empresarial

Team Size

6	Number of Team Members (full-time– equivalent people including developers, quality assurance personnel, and first-level management)

Requirements

#	Score	Question
1	**3**	Does the project have a clear, unambiguous vision statement or mission statement?
2	**3**	Do all team members believe the vision is realistic?
3	**3**	Does the project have a business case that details the business benefit and how the benefit will be measured?
4	**2**	Does the project have a user interface prototype that realistically and vividly demonstrates the functionality that the actual system will have?
5	**3**	Does the project have a detailed, written specification of what the software is supposed to do?
6	**2**	Did the project team interview people who will actually use the software (end users) early in the project and continue to involve them throughout the project?
7	**3**	Does the project have a detailed, written Software Development Plan?
8	**3**	Does the project's task list include creation of an installation program, conversion of data from previous versions of the system, integration with third-party software, meetings with the customer, and other "minor" tasks?
9	**3**	Were the schedule and budget estimates officially updated at the end of the most recently completed phase?
10	**2**	Does the project have detailed, written architecture and design documents?
11	**3**	Does the project have a detailed, written Quality Assurance Plan that requires design and code reviews in addition to system testing?
12	**3**	Does the project have a detailed Staged Delivery Plan for the software, which describes the stages in which the software will be implemented and delivered?
13	**3**	Does the project's plan include time for holidays, vacation days, sick days, and ongoing training, and are resources allocated at less than 100 percent?
14	**3**	Was the project plan, including the schedule, approved by the development team, the quality assurance team, and the technical writing team—in other words, the people responsible for doing the work?

Project Control

#	Score	Question
15	**3**	Has a single key executive who has decision-making authority been made responsible for the project, and does the project have that person's active support?
16	**3**	Does the project manager's workload allow him or her to devote an adequate amount of time to the project?
17	**3**	Does the project have well-defined, detailed milestones ("binary milestones") that are considered to be either 100 percent done or 100 percent not done?
18	**3**	Can a project stakeholder easily find out which of these binary milestones have been completed?
19	**3**	Does the project have a feedback channel by which project members can anonymously report problems to their own managers and upper managers?

20	**3**	Does the project have a written plan for controlling changes to the software's specification?
21	**3**	Does the project have a Change Control Board that has final authority to accept or reject proposed changes?
22	**3**	Are planning materials and status information for the project—including effort and schedule estimates, task assignments, and progress compared to the plan thus far—available to every team member?
23	**3**	Is all source code placed under automated revision control?
24	**3**	Does the project environment include the basic tools needed to complete the project, including defect tracking software, source code control, and project management software?

Risk Management

25	**3**	Does the project plan articulate a list of current risks to the project? Has the list been updated recently?
26	**3**	Does the project have a project risk officer who is responsible for identifying emerging risks to the project?
27	**3**	If the project uses subcontractors, does it have a plan for managing each subcontract organization and a single person in charge of each one? (Give the project full score if it doesn't use subcontractors.)

Personnel

28	**3**	Does the project team have all the technical expertise needed to complete the project?
29	**2**	Does the project team have expertise with the business environment in which the software will operate?
30	**3**	Does the project have a technical leader capable of leading the project successfully?
31	**2**	Are there enough people to do all the work required?
32	**2**	Does everyone work well together?
33	**2**	Is each person committed to the project?

Results

92	**Preliminary score**
115	**Final score (preliminary score adjusted for team size)**

Chance of Success: **Outstanding**

Evaluation: A project with this score is virtually guaranteed to succeed in all respects, meeting its schedule, budget, quality, and other targets. In terms of Chapter 1's project needs hierarchy, such a project is fully "self-actualized."

Validación de Proyectos de Sobrevivencia
Administración de Proyectos de Sistemas de Información

Data

0	**Poor**	A project with this score has significant weaknesses in the major areas of requirements, planning, project control, risk management, and personnel. The primary concern of a project in this category should be whether it will finish at all.
40	**Fair**	This score is typical. A project with this score will likely experience high stress and shaky team dynamics, and the software will ultimately be delivered with less functionality than desired at greater cost and with a longer schedule. This kind of project stands to experience the greatest benefit from applying the plan described in Software Project Survival Guide.
60	**Good**	A score in this range represents a better-than-average level of software development effectiveness. Such a project stands a fighting chance of meeting either its schedule or its budget target, but it probably won't meet both.
80	**Excellent**	A project at this level is performing much better than average. Such a project has a high probability of delivering its software close to its schedule, budget, and quality targets.
90	**Outstanding**	A project with this score is virtually guaranteed to succeed in all respects, meeting its schedule, budget, quality, and other targets. In terms of Chapter 1's project needs hierarchy, such a project is fully "self-actualized."