A to Z of software project management

GLOSSARY OF TERMS

By Justinmind, the enterprise software prototyping tool.

A to Z of software project management

Glossary of terms

Written by Emily Grace Adiseshiah & Cassandra Naji from Justinmind, the enterprise software prototyping tool

> *"Getting things done isn't just about shipping a product, or checking off items on a list, or even about marking a project as "Done." Getting things done is a process: it's a way of thinking that involves planning, execution, iteration, and reflection."*
>
> Lyndi Thompson,
> Demand Marketing at Tableau.

Smart, efficient and innovative digital solutions in the form of applications and software are changing the way we live our lives. These solutions would not be feasible if not for the time and effort put in by those who **manage** how they are defined, developed and deployed. As the software development process evolves, so do the roles involved.

However, efficient management of software must always be a software organization's number one priority.

With constant updates to software project management strategy, it can be a daunting task to unscramble the jargon: both the old and gold, and the latest terminology. From traditional software project management to waterfall to agile, our glossary provides the key to software project management terms.

INDEX

INDEX

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

GLOSSARY OF SOFTWARE PROJECT MANAGER TERMS

A

AGILE PROJECT MANAGEMENT

Agile is an iterative approach to managing the activities of engineering and information technology in a highly flexible and interactive framework. Software project managers who help organize development projects have taken on this approach with very positive results, which include improved product quality, increased quality control and faster ROI (return on investment). Agile project management aims to make the processes of software creation and delivery more efficient for internal teams and clients.

Project management is treated differently in Agile. The responsibilities of the project manager (process focus, task allocation, managing issues and dependencies, requirements management and prioritization, procurement and **risk management**) are typically divided up and covered by the Agile roles: Scrum Master, Development Team and Product Owner.

Read more on the agile management project cycle for wireframing here »

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
hU
V
W
X
Y
Z

A

APPLICATION DEVELOPMENT LIFE CYCLE

As IBM has it: “An application is a collection of programs that satisfies certain specific requirements (resolves certain problems)”.

The application development life cycle encompasses analysis and gathering requirements, design, programming and user and system test iterations, production and maintenance of an application. It is more extensive than the Software Development Life Cycle, which does not follow on to when the software is no longer in use.

The project manager of application / software development is responsible for keeping the organization’s management informed of the project’s progress, including requirements elicitation, definition and documentation.

B

BACKLOG

Prioritized wish list from which agile **sprint** tasks are selected.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

B

BASECAMP

Basecamp is a web-based project management tool with a Ruby on Rails framework, launched in 1999. In 2004, the company's focus shifted from web design to web application development, and now it exclusively deals in web applications. The tool helps users create to-do lists, manage **milestones**, network via forums, share files, and organize teams and tasks with the time tracking feature.

Read up on Justinmind's Q&A with Basecamp's UI Designer Jonas Downey and how he uses prototyping to help him manage his projects and design awesome user interfaces.

BASELINE

A project's baseline is how we measure performance – essentially the standard to measure performance against. A project's baseline is defined as its original scope, cost and **schedule** and is what the project manager will use to compare different stages of the project against.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

B

BUDGET

The combined cost of all planned activity and resources, expenses and revenues, the budget drives project funding and provides project cost control. It is important to manage a project's budget carefully. An over-budget project can be damaging to the reputation of the organization, no matter the quality or speed at which the project is delivered. Preventing cost overruns is a task of the project manager, and has been considered a real challenge for software project managers in particular. In an industry as fast-moving as that of software and application development, project managers must also keep pace in order to prevent project pitfalls, such as late deliveries, diminished quality or over-run budgets. Realistic budgeting, regular forecasting and informative **communication** between the software project manager and the rest of the team is key to reducing the risk of cost overruns.

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

B

BUSINESS CASE

A business case records the reasoning for starting a project or task. Though typically documented, it can also be agreed upon verbally or presented among stakeholders. The purpose of the business case is to establish the 'business need' that the project is fulfilling by consuming the determined resources. In software development, an upgrade is to improve performance or provide maintenance for something not working correctly. The business case here is to enhance the user experience, or satisfaction of the customer through optimizing processes and reducing costs.

Although it is not typically the role of the project manager to build the business case, they should consider it as a part of the overall landscape of the project and help them understand the wider picture.

C

CASE STUDY

The case study is a study of the development of a single or succession of instance(s) or event(s) over a period of time. It describes the behavior or the individual, group or situation through looking at events, collecting data, analyzing information, and then reporting the results or takeaway.

Case study results can help the project manager to determine how to proceed in future cases.

Check out our eBook for some examples of prototyping case studies. >>

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

C

CHANGE CONTROL

Change control is the process by which changes (both intentional and unintentional), updates and improvements to a product or software are introduced in a controlled and measured way. This includes ensuring that faults are not introduced at any point throughout the development life cycle and that other users' (team or client) changes are not undone. The process intends to **reduce rework** by avoiding the need for additional changes further down the line, with minimal disruption to production.

Change control is an essential aspect of any software production system, from installation of new operating systems to upgrades and changes to infrastructure. Change control and change management is an "organizational constant", and is a challenge for everyone, not just those directly affected by it.

It's the project manager's role to **document and discuss project expectations and changes** with team, client and stakeholders. If a change is suggested by either a team member or client, it is the project manager who will carry out the change process upon review and approval of the suggestion. This will include considering the overall effect on budget, time impact and additional risks associated, communicating the suggested change to the decision-making authority, overseeing implementation of the change, and closing up the process with a Change Log.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

C

COMMUNICATION

More effective communication = Better project management

Effective communication is the number one soft skill for a project manager, and perhaps one of the hardest to achieve. According to the Project Management Institute, around 90% of a project's life cycle is spent on communication by the project manager with the rest of the team and the client. Each step in a project requires a discussion: What tasks does this step require? Who will work in them? How much of our budget will we spend? Are we on track? All of these questions and their respective answers are funneled through the project manager.

As Brett Harned puts it, "Project management is more than tools and process. It's also about also people and working with them to produce projects that meet or exceed requirements."

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

D

DEPENDENCIES

Dependencies describe the relationship between a set of tasks and help to determine the order in which these tasks should be carried out within a project. There are four types of dependency relationships:

Finish to Start (FS): (B) cannot start before (A) is finished. Budget (B) cannot be approved until requirements (A) are confirmed.

Finish to Finish (FF): (B)v cannot finish before (A) is finished, but (B) can finish any time after (A) finishes. Development (B) cannot finish before **wireframe** (A) is complete.

Start to Start (SS): (B) cannot start before (A) starts. Implementing interaction in UI (B) cannot begin until the UI design (A) has begun.

Start to Finish (SF): (B) cannot finish before (A) starts. Invoicing the client (B) cannot finish before the project (A) kicks off.

In project management, the most common dependency relationship and the default type of dependency that a project will use is a finish-to-start relationship. The least common is the start-to-finish relationship.

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

D

DELPHI TECHNIQUE

Delphi is a communication technique that helps project managers to forecast the likelihood and outcome of future events in a project. It involves a panel or group of experts exchanging estimates, views and assumptions to give a forecast report a facilitator. The facilitator will then provide a second round of review, and then issue a second report. It is an iterative process, continuing until each individual expert reaches a consensus. For each round, each expert revises earlier forecasts, privately. The anonymity is important because it avoids the experts being influenced by each other, allows them to concentrate on revising the forecasts without distraction, and ultimately identify risks and opportunities within the project. The mean or median scores of the final rounds determine the results.

A famous example of the Delphi Technique is the Horizon Project. Here, educational ‘futurists’ determine possible technological advancements that could come about in education within the next few years.

E

EPIC (AGILE)

In Agile, an epic is a set of user stories – a really big user story that captures a large body of work and can be broken down into a number of smaller **user stories**. In software to be released to the client, an epic can span multiple agile project sprints, **versions** and projects. They allow teams to create hierarchy and structure and can change in scope (a concept in line with agile development).

E

ESTIMATION

Project estimates help project managers to plan out cost, timelines, and level of effort needed to make a project successful. They also enable the team to prioritize tasks, based on a number of variables including level of urgency, size and resources required.

Although estimates are a useful way to map out how a project will occur, it can be daunting task. Why? Because the project manager will only know the ins and outs of the above variables once the project is complete. In order to create accurate, workable estimates, the project manager needs to know everything about their team and their processes, the deliverables and planned tasks. Questions need to be put to the team, client and perhaps the user to avoid gaps in the process.

With the rise of Agile, we're seeing project estimates shift into user story estimates. This is about shifting the focus from writing about requirements to talking about them – an arguably more efficient approach to project management.

Read up on how Justinmind's Product VP manages the software team's sprint backlog here. »

> ***"The Agile Project Management principles and framework encourage learning and adapting as an integral part of delivering value to customers."***
>
> Jim Highsmith, Agile Project Management: Creating Innovative Products

F

FUNCTIONAL REQUIREMENTS

A subunit of software requirements specification, functional requirements define the function and activities of a system, its components or subsystems. A function is as a set of inputs, the behavior, and outputs of the system.

Functional requirements are determined by the type of software, target users and type of system where the software will be used and typically contain a unique name and number, a brief summary, and a rationale (descriptions of data, **operations**, workflows, system reports). Examples of functional requirements include business rules, authentication, external interfaces and historical data.

The client is ultimately responsible for specifying requirements, but the project manager or business analyst typically acts as the liaison between the team and client.

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

G

GANTT, HENRY

Henry Laurence Gantt was an American mechanical engineer and management consultant who is best known for his work in the development of scientific management. He created the **Gantt chart** in the 1910s, a project management tool that provides visual methods of scheduling both time and resources for projects. The chart illustrates the start and finish dates of the elements within a project.

Although outdated, the Gantt chart was a pioneering advance in charting techniques.

H

HERMES

Hermes is a project management method that supports the driving, management, and execution of projects through a series of **scenarios, modules, phases and milestones, roles and tasks and results.** The HERMES scenarios include but are not limited to:

- Customized IT application: developing application or software that meets a specific need of a department and integrating it into the organizational infrastructure.

- IT Application Upgrade: upgrading existing application or software.

- Organizational Adjustments: adapting the organizational structure and processes of an organizational unit.

Depending on the nature of the project, scenarios can be adapted. The project manager is typically responsible for initiating the HERMES method within an organization.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

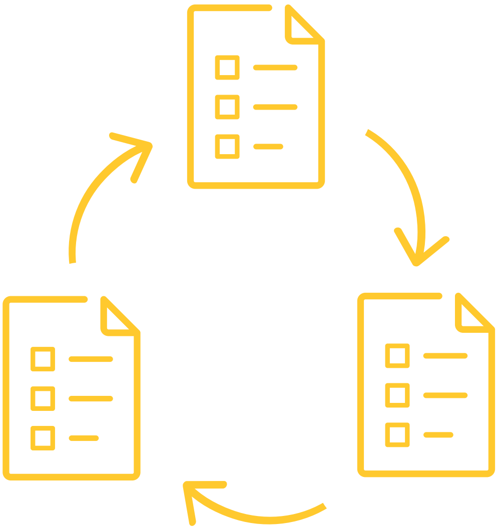
INCREMENTAL DEVELOPMENT

The incremental method involves designing, implementing and testing software cumulatively until the software is complete. In contrast to the cyclic approach to development that iterative design provides, incremental development uses a set number of steps along a linear path of progression. The **Waterfall model** is a traditional incremental development approach.

ITERATION

A short time frame within which to deliver a software feature or set of software features.   In agile product development, there are often **multiple iterations**, each with an end review stage to assess the changes made.

Read up on the iterative design process using Justinmind here. »



A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

I

ITERATIVE DESIGN & DEVELOPMENT

Iterative development is an approach to application and software development that rolls out development in phases. It is a cyclic process of prototyping, testing and refining software to improve its quality and functionality as much as possible prior to delivery.

The phases of iterative development are:

• The requirements phase, in which requirements are defined with the client.

• The design phase, in which the software to meet the requirements is designed, often with a series of wireframes, mockups and interactive, functional prototypes.

• Implementation and code phase, in which the software is coded, implemented and tested. And finally,

• The review phase, in which the software is analyzed against the requirements and rounds of changes (iterations) are performed as needed.

The iterations require that this approach has no set number of steps. This development method is quite different the non-iterative approach in terms of the nature of milestones, the dependencies involved, the resources required and the team interaction and dynamic. A steady hand is required of the project manager to balance all of the responsibilities.

Read more on iterative design in our case study here. »

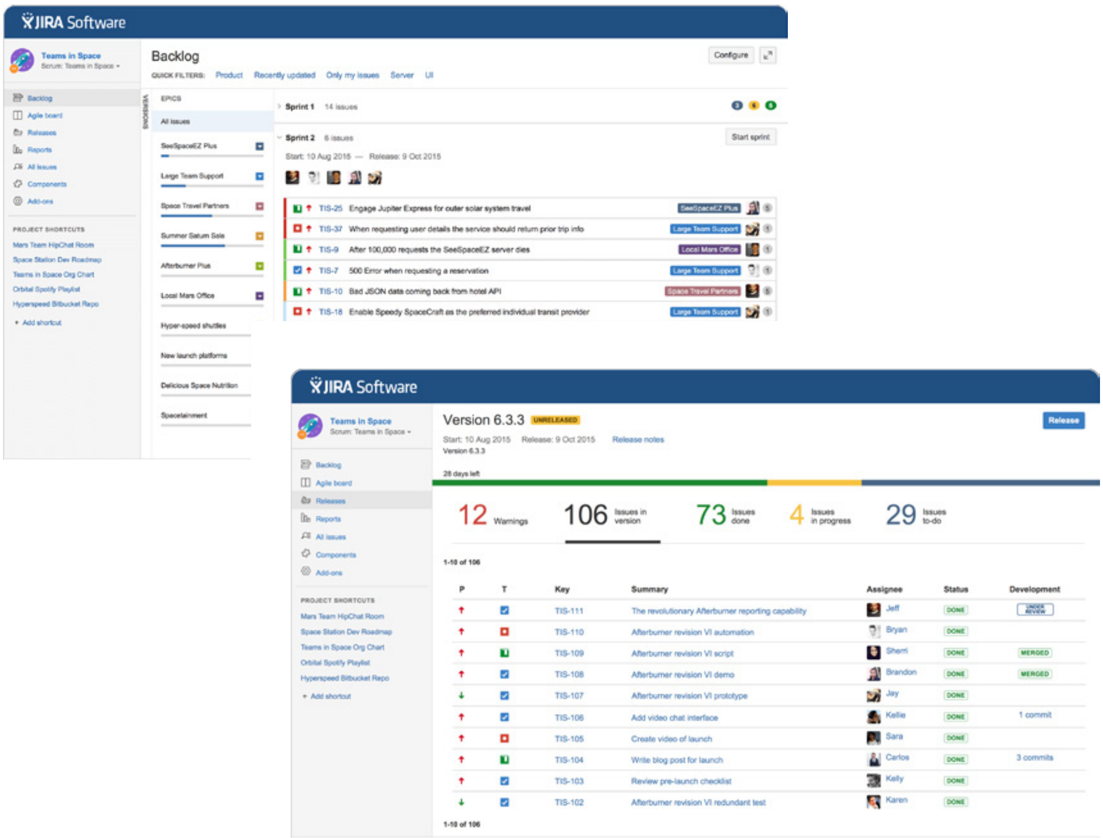
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

J

JIRA

Atlassian JIRA is a project management tool that helps project managers (and the rest of the team) to capture, assign, and set priorities to their projects. Users can manage the entire process of application and software development by creating, managing and tracking sprints, as well as keeping a backlog of additional tasks. A favorite of design and development teams, JIRA allows users to add project milestones using versions to keep the project on track, perform project reporting to help measure goals, and customize issues to fit the needs of each team.

Project managers can visualize team workflows and the issue lifecycle for each project phase.



Key	Summary	Epic
TIS-25	Engage Jupiter Express for outer solar system travel	SeeSpaceEZ Plus
TIS-37	When requesting user details the service should return prior trip info	Large Team Support
TIS-9	After 100,000 requests the SeeSpaceEZ server dies	Local Mars Office
TIS-7	500 Error when requesting a reservation	Large Team Support
TIS-10	Bad JSON data coming back from hotel API	Space Travel Partners
TIS-18	Enable Speedy SpaceCraft as the preferred individual transit provider	Large Team Support

1-10 of 106

Key	Summary	Assignee	Status	Development
TIS-111	The revolutionary Afterburner reporting capability	Jeff	DONE	UNDER REVIEW
TIS-110	Afterburner revision VI automation	Bryan	DONE	
TIS-109	Afterburner revision VI script	Sherri	DONE	MERGED
TIS-108	Afterburner revision VI demo	Brandon	DONE	MERGED
TIS-107	Afterburner revision VI prototype	Jay	DONE	
TIS-106	Add video chat interface	Kellie	DONE	1 commit
TIS-105	Create video of launch	Sara	DONE	
TIS-104	Write blog post for launch	Carlos	DONE	3 commits
TIS-103	Review pre-launch checklist	Kelly	DONE	
TIS-102	Afterburner revision VI redundant test	Karen	DONE	

1-10 of 106

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

J

JUSTINMIND

Didn't think we'd leave ourselves out, did you? Justinmind prototyping tool offers an intuitive approach to requirements definition and management. Create requirements and customize them to the needs of each client, or team. Our tool is also integrated with **Atlassian JIRA** and Microsoft TFS to avoid rework. Simply connect your Justinmind with either (or both) project management tool(s) to view and modify issues and tasks within the prototyping tool. You can then export them back to JIRA and TFS and view the updates.

DOWNLOAD JUSTINMIND

K

KANBAN

Kanban is an agile project management and scheduling framework. It is a streamlined approach to project management that aims to avoid waste within the manufacturing process. Japanese for "visual signal" in the literal sense of the word, the English-term "queue limiter" most closely captures its real meaning. Kanban has its roots in Lean production, originally put into practice by car manufacturer Toyota in the 1940s, but has since become an important scheduling system in other industries, including software development.

Learn more about Kanban here. »

K

KICKOFF MEETING

Any successful project needs a great kickoff meeting. Once the budget is approved, the team and client will sit down to their first project meeting. This is when the tone of the project is set, first impressions are made and estimations start being made. Project objectives, tools, schedule and roles are defined at this stage, as well as potential risks identified. This is also the moment when the workflow is outlined, as well as best practices.

L

LEAN MANUFACTURING

"The best way to do MORE could be by doing NOTHING" Cameron Ford, The Lean Post

Learn manufacturing is a systematic approach to production that intends to improve quality, reduce lead time, reduce total costs and above all prevent activity that does not add value from the client's perspective (waste).

According to research conducted by the Lean Enterprise Research Centre (LERC), an immense 60% of production activities in a typical manufacturing operation add no value at all for the customer.

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

M

MILESTONE

Milestones are used as signal posts or reference points to track specific points in a project timeline and measure the progress of a project toward its ultimate goal. These points can signal project start and end date, call for external review, additional input and/or budget checks or updates, or document a major deliverable. In many instances, milestones do not impact project duration.

MOCKUP

In software development, the mockup is a model of a design or device. Mockups can range from simple paper sketches to semi-functional user interface models built with a software development tool. Whether made with paper or with digital tools, the mockup is typically a static, visual representation of a website or mobile application's user interface or screen. It intends to help the viewer visualize the basic layout, design and content as well as demonstrate the essential functionality of the user interface before any interaction is added.

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

N

NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements define how a software system will do what it has been built to do – operations rather than behavior or functions, which relate to **functional requirements**.

Examples of non-functional requirements include performance, scalability, reliability and security.

O

OPERATIONS

Operations are ongoing, recurring functions that contribute to the running of the organization. They are permanent efforts that deliver continual production.

Operations management is often contrasted with project management. Project management intends to improve processes through temporary projects that are designed to respond immediately to external obstacles. In contrast, the goal of operations management is to increase effectiveness, reduce costs and gain competitive advantage. Organizations use projects to adapt operations to meet business needs and respond to changing markets.

> ***"If you can't describe what you are doing as a process, you don't know what you are doing"***
>
> W. Edwards Deming

P

PROJECT MANAGEMENT

According to the Project Management Institute, project management is "the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements."

PROTOTYPE

A prototype is a first or early model of a device, product or service from which more advanced, exact forms are produced. In IT systems, a web or UI prototype helps teams to visualize how their interface is going to look and feel once developed.

Prototyping tools can help project managers improve collaboration efforts by consolidating teams on one platform to work simultaneously on project tasks.

DOWNLOAD JUSTINMIND

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

P

PRIORITIZATION

Prioritization is the process of managing the relative importance and urgency of different tasks or requirements in order to make production more efficient with the fewest resources consumed – ultimately to maximize stakeholder ROI (return on investment). In software development, prioritization is used to help teams decide which tasks to work on in each phase of the project.

In Agile, prioritization aims to help teams develop software which is both high-quality and high-value by working on the highest-priority tasks first. The **Scrum** approach to managing requirements involves a backlog of tasks which is stacked according to priority, calculated by **estimates**.

Q

QUALITY ASSURANCE

Quality assurance is how organizations prevent mistakes, errors and/or defects in production or during the process of delivery. Standards, documents that provide requirements and guidelines on how to ensure products, processes and services are fit for purpose, are often included in the quality assurance processes of organizations. For instance, international standard ISO 9000, which deals with the fundamentals of quality management systems, "helps organizations ensure that they meet the needs of customers and other stakeholders while meeting statutory and regulatory requirements related to a product or program".

Software quality assurance (SQA) involves monitoring and auditing the software development process to ensure the highest possible quality throughout.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

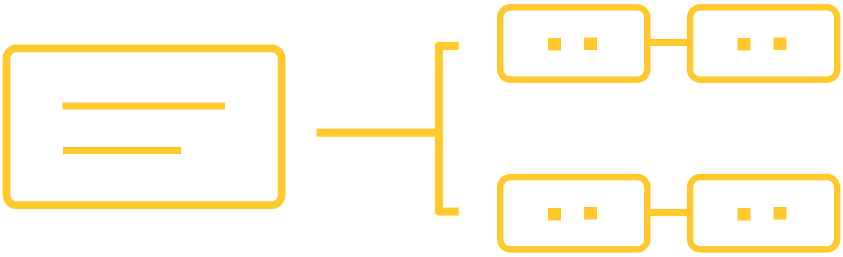
R

REQUIREMENT CREEP

Requirement creep, also known as scope, function, feature creep, (and also kitchen sink syndrome!), refers to continuous changes to requirements in a project's scope any time after **kickoff**. Creep can be harmful to a project and organization as it inflates resources and costs, without raising the price. Additionally, once an organization accepts creep from a client, it sets a precedent for future projects and can damage the organization's reputation.

Modern project management methodologies have ways to avoid the occurrence of requirement creep. In the Waterfall methodology, **change control** prevents changes from being requested upon project kickoff by having everyone involved in the project (organization plus client) agrees that the scope won't be modified. However, some argue that this is too inflexible an approach, as requirements often need to be modified in software development to allow for unexpected emergencies, such as bugs during testing.

Agile's approach to preventing requirement creep is more flexible, allowing for some planning to take place in the moment, rather than prior to kick off. The adaptability means that changes can be implemented without penalty.



A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

R

RISK MANAGEMENT

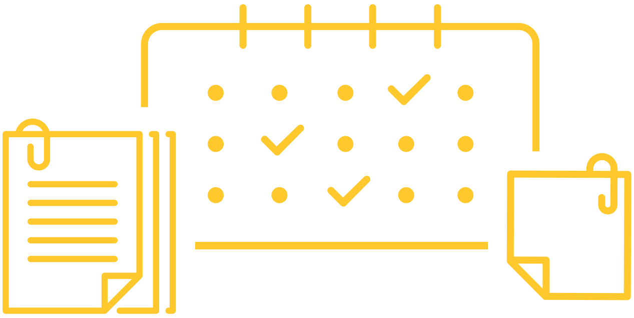
Risk management is the process of identifying and assessing potential risks and then applying and prioritizing resources to minimize and/or control the probability of their occurrence.

As software becomes the go-to for completing business-critical operations, software risk management is essential. System failures result in loss of revenue, delays in or freezing of operations, and security breaches. Establishing requirements early on, defining clear escalation paths and complying with quality control efforts through development will help to reduce the risk of system failures.

S

SCHEDULES

A schedule is a list of what needs to be done, which resources are required, and what the deliverables are in a project. Project managers typically use project schedules in project planning, once **estimates**, **budget, dependencies** and resource allocation are confirmed. With the adoption of agile across a number of industries, and having a big presence in software development organizations, it's become common practice to combine traditional project management with agile project management tools such as **Atlassian JIRA**.



A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

S

SCRUM

Scrum is an agile scheduling and managing framework for software development (see **Kanban** here). Development teams work as a unit to product software in a flexible, iterative system.

As described by Springtimesoft, Scrum works like this:

- The Scrum Product Owner creates product **backlog**.
- The **sprint** planning takes place. The Scrum Team will pick the highest-priority tasks from the backlog and decide how to implement them (who, when, what).
- The Scrum Team works on the set of tasks within a pre-defined time frame (sprint) to complete the work its work and meets each day to assess its progress (daily Scrum).
- The Scrum Master's job is to keep track of the team's progress and keep members focused on the sprint's goals.
- At the end of the sprint, the work should be ready for the next phase: testing, rounds of review, or shippable for delivery.
- The sprint ends with a sprint review and a retrospective meeting to determine and discuss what went well, what could be improved, and what should be scrapped. Sometimes best practices will be created from the retrospective.
- Once the next sprint begins, the Scrum team returns to the product backlog to select the tasks to be performed.

A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

S

SPRINT

In agile product development, a sprint defines the time frame (usually two to four weeks) within which a set of tasks should be completed and made ready for review or delivery.

Each sprint begins with a meeting between the Product Owner and the Scrum team to plan which tasks will be included, by who and the time needed.

T

TEAM COLLABORATION TOOLS

Collaboration (and communication) is crucial to the project management system. The project manager is the go-between for different teams working on software, and spends a great deal of their day passing messages back and forth. It's hard enough to get everyone on the same page, without the added complication of reaching teams who are located in different departments, or even in different countries. Team collaboration tools help to manage team workflows and task-management, and encourage communication. For instance, Asana is primarily web-based tool that is comprised of projects and teams. Each project has a task list and team members can add and delete tasks and comment in the list.

> ***"A project manager is supposed to be a leader, a facilitator, an advisor and an advocate, not only for the project, but also for the team, the business and the cause"***
>
> Keren Nimmo, The Seven Deadly Sins of Project Management

U

USER STORY

In agile product development, a user story is a short and simple description of one or more of the features of a software system. It is told from the perspective of the person or group who will benefit from the new functionality, design or interaction – usually the user.

V

VERSION

A version describes a specific point in time (or unique state) for a project that is used in software development to help teams schedule and manage different software releases an updates.

W

WATERFALL METHODOLOGY

Waterfall is a non-iterative approach to software development. Progress in waterfall is a linear and discrete, flowing steadily downwards through the phases of requirements gathering and analysis, system design, implementation, testing, deployment and maintenance.

In waterfall, requirements are very well documented and are fixed, as are deadlines. Product definition is stable and inflexible, in contrast to the adaptable approach that agile adopts. This methodology allows departmentalization and more control over variables in a project. The drawback of waterfall is that it offers little room for reflection or retrospective, where agile excels.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

W

WIREFRAME

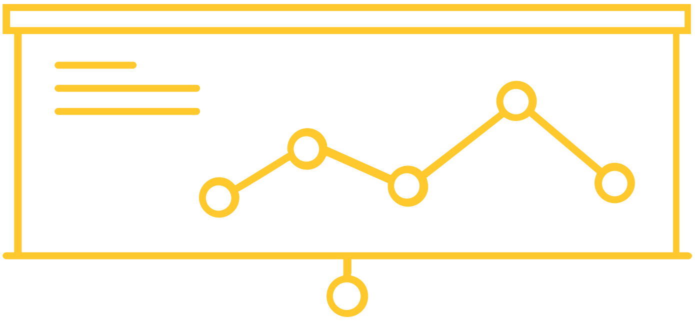
A wireframe is a skeletal (paper or digital) model that allows you to define the information hierarchy or your design and plan the layout before adding interactive, functional details to a high-fidelity **prototype**.

DOWNLOAD JUSTINMIND

X

X-BAR CONTROL CHARTS

According to Bright Hub Project Management, X-bar/R control charts are two control charts in one: a chart of the average (x-bar) over time, plus a chart of the range (R) over time. Project managers may use these control charts to that pre-production is in order.



> *"Prototyping helps project managers leverage resources to deliver top quality software development projects"*
>
> Justinmind

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Y

YES, YOU CAN!

Ok, we cheated. This isn't actually a project management term. But who cares? Here's some PM cheer to get you through Z.

Z

(SPRINT) ZERO

Sprint zero refers to the initial sprint of a software project, which teams use to prepare their product backlog, working environment, and begin to prioritize features or user stories with estimates.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

It is the project manager, or the equivalent of in Agile or Waterfall, who is primarily responsible for the successful delivery of the application or software. Organizing the inevitable chaos, managing the risks and bringing teams together, these individuals are often called upon to go above and beyond in the name of delivering on commitments – more so than ever in software projects where things can be somewhat uncertain throughout. Projects have the potential to fail, this much we know. But it is the role of the project manager to ensure that teams learn from mistakes, take away the lesson, and fail forward.

JUST
INMIND

www.justinmind.com

jim.info@justinmind.com

180 Sansome St. Fl. 2
San Francisco
CA 94104, USA