Chapter 22 – Project Management

Topics covered

- ✧ Risk management
- ✧ Managing people
- ✧ Teamwork

Software project management

- ✧ Concerned with activities involved in ensuring that software is delivered on time and on schedule and in accordance with the requirements of the organisations developing and procuring the software.
- ✧ Project management is needed because software development is always subject to budget and schedule constraints that are set by the organisation developing the software.

Success criteria

- Deliver the software to the customer at the agreed time.
- Keep overall costs within budget.
- Deliver software that meets the customer’s expectations.
- Maintain a coherent and well-functioning development team.

Software management distinctions

- The product is intangible.
  - Software cannot be seen or touched. Software project managers cannot see progress by simply looking at the artefact that is being constructed.
- Many software projects are 'one-off' projects.
  - Large software projects are usually different in some ways from previous projects. Even managers who have lots of previous experience may find it difficult to anticipate problems.
- Software processes are variable and organization specific.
  - We still cannot reliably predict when a particular software process is likely to lead to development problems.

Factors influencing project management

- Company size
- Software customers
- Software size
- Software type
- Organizational culture
- Software development processes
- These factors mean that project managers in different organizations may work in quite different ways.

Universal management activities

- ✧ *Project planning*
  - Project managers are responsible for planning. estimating and scheduling project development and assigning people to tasks.
- ✧ *Risk management*
  - Project managers assess the risks that may affect a project, monitor these risks and take action when problems arise.
- ✧ *People management*
  - Project managers have to choose people for their team and establish ways of working that leads to effective team performance.

Management activities

✧ *Reporting*

- Project managers are usually responsible for reporting on the progress of a project to customers and to the managers of the company developing the software.

✧ *Proposal writing*

- The first stage in a software project may involve writing a proposal to win a contract to carry out an item of work. The proposal describes the objectives of the project and how it will be carried out.

Risk management

Risk management

- ✧ Risk management is concerned with identifying risks and drawing up plans to minimise their effect on a project.
- ✧ Software risk management is important because of the inherent uncertainties in software development:
  - loosely defined requirements,
  - requirements changes due to changes in customer needs,
  - difficulties in estimating the time and resources required for software development, and
  - differences in individual skills.
- ✧ You have to anticipate risks, understand the impact of these risks on the project, the product and the business, and take steps to avoid these risks.

Risk classification

- ✧ One dimension of risk classification says what is affected by the risk:
  - *Project risks* affect schedule or resources;
  - *Product risks* affect the quality or performance of the software being developed;
  - *Business risks* affect the organisation developing or procuring the software.
- ✧ Other one indicates the type of risk (e.g., technical, organizational)

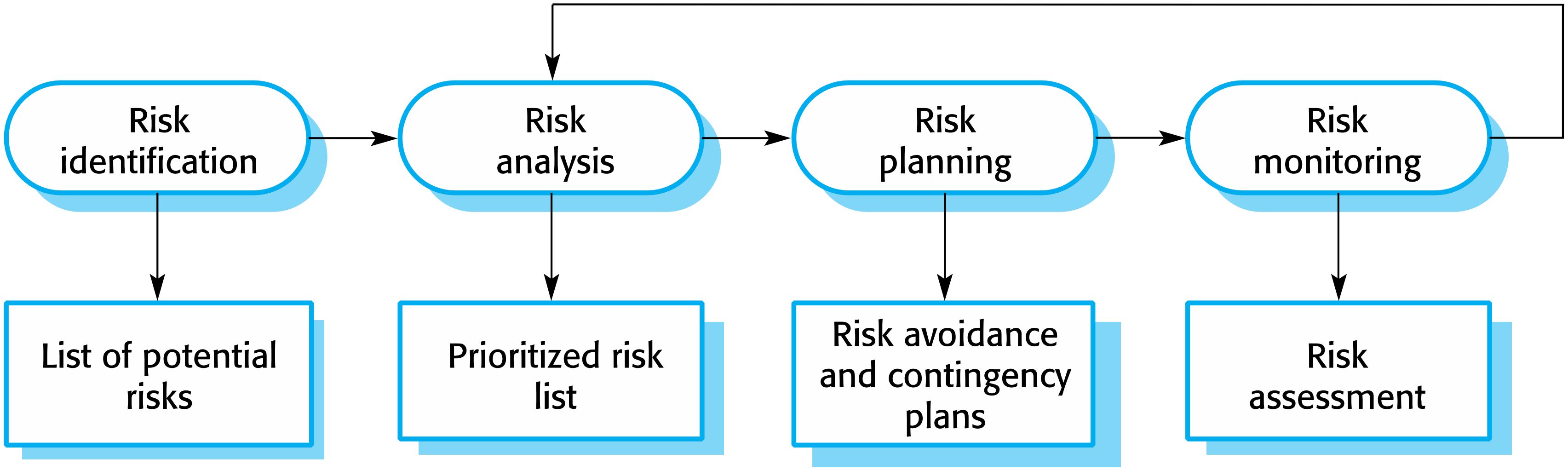
Examples of project, product, and business risks

Risk	Affects	Description
Staff turnover	Project	Experienced staff will leave the project before it is finished.
Management change	Project	There will be a change of organizational management with different priorities.
Hardware unavailability	Project	Hardware that is essential for the project will not be delivered on schedule.
Requirements change	Project and product	There will be a larger number of changes to the requirements than anticipated.
Specification delays	Project and product	Specifications of essential interfaces are not available on schedule.
Size underestimate	Project and product	The size of the system has been underestimated.
CASE tool underperformance	Product	CASE tools, which support the project, do not perform as anticipated.
Technology change	Business	The underlying technology on which the system is built is superseded by new technology.
Product competition	Business	A competitive product is marketed before the system is completed.

The risk management process

- Risk identification
  - Identify project, product and business risks;
- Risk analysis
  - Assess the likelihood and consequences of these risks;
- Risk planning
  - Draw up plans to avoid or minimise the effects of the risk;
- Risk monitoring
  - Monitor the risks throughout the project;

The risk management process



Risk identification → List of potential risks

Risk analysis → Prioritized risk list

Risk planning → Risk avoidance and contingency plans

Risk monitoring → Risk assessment

Risk identification

- ✧ May be a team process or based on the individual project manager’s experience.
- ✧ A checklist of common risks may be used to identify risks in a project
  - Technology risks.
  - Organizational risks.
  - People risks.
  - Requirements risks.
  - Estimation risks.

Examples of different risk types

Risk type	Possible risks
Estimation	The time required to develop the software is underestimated. (12) The rate of defect repair is underestimated. (13) The size of the software is underestimated. (14)
Organizational	The organization is restructured so that different management is responsible for the project. (6) Organizational financial problems force reductions in the project budget. (7)
People	It is impossible to recruit staff with the skills required. (3) Key staff are ill and unavailable at critical times. (4) Required training for staff is not available. (5)
Requirements	Changes to requirements that require major design rework are proposed. (10) Customers fail to understand the impact of requirements changes. (11)
Technology	The database used in the system cannot process as many transactions per second as expected. (1) Reusable software components contain defects that mean they cannot be reused as planned. (2)
Tools	The code generated by software code generation tools is inefficient. (8) Software tools cannot work together in an integrated way. (9)

Risk analysis

- Assess probability and seriousness of each risk.
- Probability may be very low, low, moderate, high or very high.
- Risk consequences might be catastrophic, serious, tolerable or insignificant.

Risk types and examples

Risk	Probability	Effects
Organizational financial problems force reductions in the project budget (7).	Low	Catastrophic
It is impossible to recruit staff with the skills required for the project (3).	High	Catastrophic
Key staff are ill at critical times in the project (4).	Moderate	Serious
Faults in reusable software components have to be repaired before these components are reused. (2).	Moderate	Serious
Changes to requirements that require major design rework are proposed (10).	Moderate	Serious
The organization is restructured so that different management are responsible for the project (6).	High	Serious
The database used in the system cannot process as many transactions per second as expected (1).	Moderate	Serious

Risk types and examples

Risk	Probability	Effects
The time required to develop the software is underestimated (12).	High	Serious
Software tools cannot be integrated (9).	High	Tolerable
Customers fail to understand the impact of requirements changes (11).	Moderate	Tolerable
Required training for staff is not available (5).	Moderate	Tolerable
The rate of defect repair is underestimated (13).	Moderate	Tolerable
The size of the software is underestimated (14).	High	Tolerable
Code generated by code generation tools is inefficient (8).	Moderate	Insignificant

Risk planning

- ✧ Consider each risk and develop a strategy to manage that risk.
- ✧ Avoidance strategies
  - ▪ The probability that the risk will arise is reduced;
- ✧ Minimization strategies
  - ▪ The impact of the risk on the project or product will be reduced;
- ✧ Contingency plans
  - ▪ If the risk arises, contingency plans are plans to deal with that risk;

What-if questions

- ✧ What if several engineers are ill at the same time?
- ✧ What if an economic downturn leads to budget cuts of 20% for the project?
- ✧ What if the performance of open-source software is inadequate and the only expert on that open source software leaves?
- ✧ What if the company that supplies and maintains software components goes out of business?
- ✧ What if the customer fails to deliver the revised requirements as predicted?

Strategies to help manage risk

Risk	Strategy
Organizational financial problems	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business and presenting reasons why cuts to the project budget would not be cost-effective.
Recruitment problems	Alert customer to potential difficulties and the possibility of delays; investigate buying-in components.
Staff illness	Reorganize team so that there is more overlap of work and people therefore understand each other’s jobs.
Defective components	Replace potentially defective components with bought-in components of known reliability.
Requirements changes	Derive traceability information to assess requirements change impact; maximize information hiding in the design.

Strategies to help manage risk

Risk	Strategy
Organizational restructuring	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business.
Database performance	Investigate the possibility of buying a higher-performance database.
Underestimated development time	Investigate buying-in components; investigate use of a program generator.

Risk monitoring

- Assess each identified risks regularly to decide whether or not it is becoming less or more probable.
- Also assess whether the effects of the risk have changed.
- Each key risk should be discussed at management progress meetings.

Risk indicators

Risk type	Potential indicators
Estimation	Failure to meet agreed schedule; failure to clear reported defects.
Organizational	Organizational gossip; lack of action by senior management.
People	Poor staff morale; poor relationships amongst team members; high staff turnover.
Requirements	Many requirements change requests; customer complaints.
Technology	Late delivery of hardware or support software; many reported technology problems.
Tools	Reluctance by team members to use tools; complaints about CASE tools; demands for higher-powered workstations.

Managing people

Managing people

- ✧ People are an organisation's most important assets.
- ✧ The tasks of a manager are essentially people-oriented. Unless there is some understanding of people, management will be unsuccessful.
- ✧ Poor people management is an important contributor to project failure.

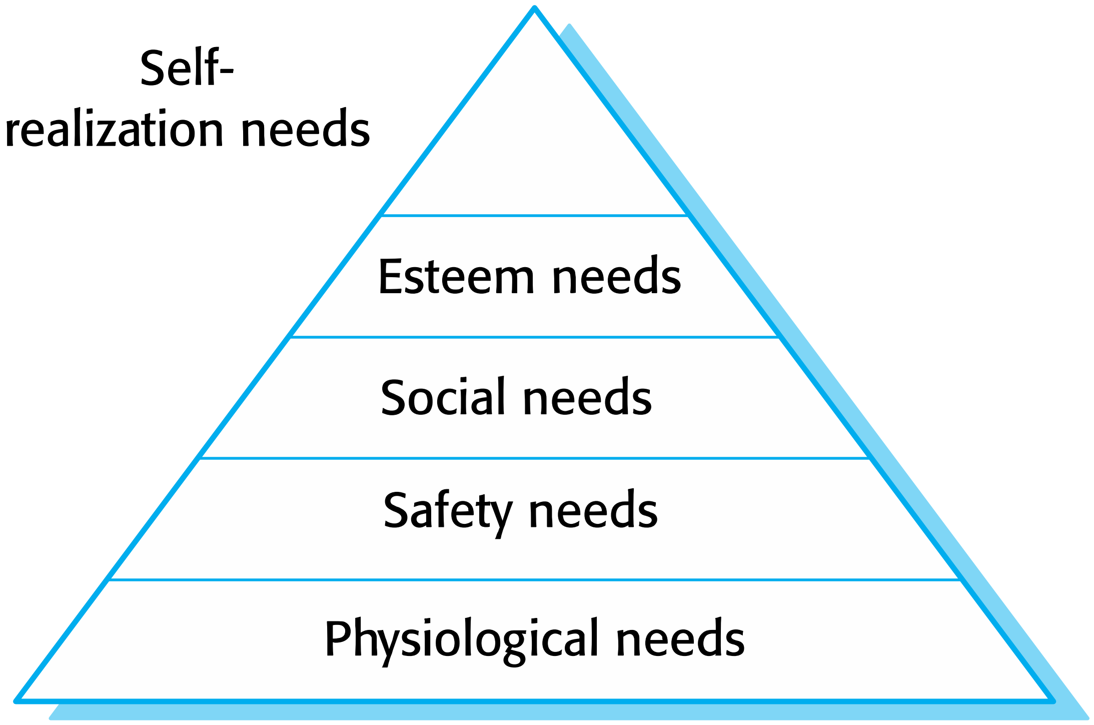
People management factors

- ✧ Consistency
  - Team members should all be treated in a comparable way without favourites or discrimination.
- ✧ Respect
  - Different team members have different skills and these differences should be respected.
- ✧ Inclusion
  - Involve all team members and make sure that people’s views are considered.
- ✧ Honesty
  - You should always be honest about what is going well and what is going badly in a project.

Motivating people

- ✧ An important role of a manager is to motivate the people working on a project.
- ✧ Motivation means organizing the work and the working environment to encourage people to work effectively.
  - If people are not motivated, they will not be interested in the work they are doing. They will work slowly, be more likely to make mistakes and will not contribute to the broader goals of the team or the organization.
- ✧ Motivation is a complex issue but it appears that there are different types of motivation based on:
  - Basic needs (e.g. food, sleep, etc.);
  - Personal needs (e.g. respect, self-esteem);
  - Social needs (e.g. to be accepted as part of a group).

Human needs hierarchy



Need satisfaction

- In software development groups, basic physiological and safety needs are not an issue.
- Social
  - Provide communal facilities;
  - Allow informal communications e.g. via social networking
- Esteem
  - Recognition of achievements;
  - Appropriate rewards.
- Self-realization
  - Training - people want to learn more;
  - Responsibility.

Personality types

- The needs hierarchy is almost certainly an over-simplification of motivation in practice.
- Motivation should also take into account different personality types:
    - Task-oriented people, who are motivated by the work they do. In software engineering.
    - Interaction-oriented people, who are motivated by the presence and actions of co-workers.
    - Self-oriented people, who are principally motivated by personal success and recognition.

Personality types

- ✧ Task-oriented.
  - The motivation for doing the work is the work itself;
- ✧ Self-oriented.
  - The work is a means to an end which is the achievement of individual goals - e.g. to get powerful, famous;
- ✧ Interaction-oriented
  - The principal motivation is the presence and actions of co-workers.

Motivation balance

- ✧ Individual motivations are made up of elements of each class.
- ✧ The balance can change depending on personal circumstances and external events.
- ✧ However, people are not just motivated by personal factors but also by being part of a group and culture.
- ✧ People go to work because they are motivated by the people that they work with.

Teamwork

Teamwork

- Most software engineering is a group activity
  - The development schedule for most non-trivial software projects is such that they cannot be completed by one person working alone.
- A good group is cohesive and has a team spirit. The people involved are motivated by the success of the group as well as by their own personal goals.
- Group interaction is a key determinant of group performance.
- Flexibility in group composition is limited
  - Managers must do the best they can with available people.

Group cohesiveness

- In a cohesive group, members consider the group to be more important than any individual in it.
- The advantages of a cohesive group are:
  - Group quality standards can be developed by the group members.
  - Team members learn from each other and get to know each other’s work; Inhibitions caused by ignorance are reduced.
  - Knowledge is shared. Continuity can be maintained if a group member leaves.
  - Refactoring and continual improvement is encouraged. Group members work collectively to deliver high quality results and fix problems, irrespective of the individuals who originally created the design or program.

The effectiveness of a team

- The people in the group
  - You need a mix of people in a project group as software development involves diverse activities such as negotiating with clients, programming, testing and documentation.
- The group organization
  - A group should be organized so that individuals can contribute to the best of their abilities and tasks can be completed as expected.
- Technical and managerial communications
  - Good communications between group members, and between the software engineering team and other project stakeholders, is essential.

Selecting group members

- ✧ A manager or team leader’s job is to create a cohesive group and organize their group so that they can work together effectively.
- ✧ This involves creating a group with the right balance of technical skills and personalities, and organizing that group so that the members work together effectively.

Assembling a team

- ✧ May not be possible to appoint the ideal people to work on a project
  - ▪ Project budget may not allow for the use of highly-paid staff;
  - ▪ Staff with the appropriate experience may not be available;
  - ▪ An organisation may wish to develop employee skills on a software project.
- ✧ Managers have to work within these constraints especially when there are shortages of trained staff.

Group composition

- ✧ Group composed of members who share the same motivation can be problematic
  - Task-oriented - everyone wants to do their own thing;
  - Self-oriented - everyone wants to be the boss;
  - Interaction-oriented - too much chatting, not enough work.
- ✧ An effective group has a balance of all types.
- ✧ This can be difficult to achieve software engineers are often task-oriented.
- ✧ Interaction-oriented people are very important as they can detect and defuse tensions that arise.

Group organization

- The way that a group is organized affects the decisions that are made by that group, the ways that information is exchanged and the interactions between the development group and external project stakeholders.
  - Key questions include:
    - Should the project manager be the technical leader of the group?
    - Who will be involved in making critical technical decisions, and how will these be made?
    - How will interactions with external stakeholders and senior company management be handled?
    - How can groups integrate people who are not co-located?
    - How can knowledge be shared across the group?

Group organization

- Small software engineering groups are usually organised informally without a rigid structure.
- For large projects, there may be a hierarchical structure where different groups are responsible for different sub-projects.
- Agile development is always based around an informal group on the principle that formal structure inhibits information exchange

Informal groups

✧ The group acts as a whole and comes to a consensus on decisions affecting the system.

✧ The group leader serves as the external interface of the group but does not allocate specific work items.

✧ Rather, work is discussed by the group as a whole and tasks are allocated according to ability and experience.

✧ This approach is successful for groups where all members are experienced and competent.

Group communications

- ✧ Good communications are essential for effective group working.
- ✧ Information must be exchanged on the status of work, design decisions and changes to previous decisions.
- ✧ Good communications also strengthens group cohesion as it promotes understanding.

Group communications

- ✧ Group size
  - The larger the group, the harder it is for people to communicate with other group members.
- ✧ Group structure
  - Communication is better in informally structured groups than in hierarchically structured groups.
- ✧ Group composition
  - Communication is better when there are different personality types in a group and when groups are mixed rather than single sex.
- ✧ The physical work environment
  - Good workplace organisation can help encourage communications.

Key points

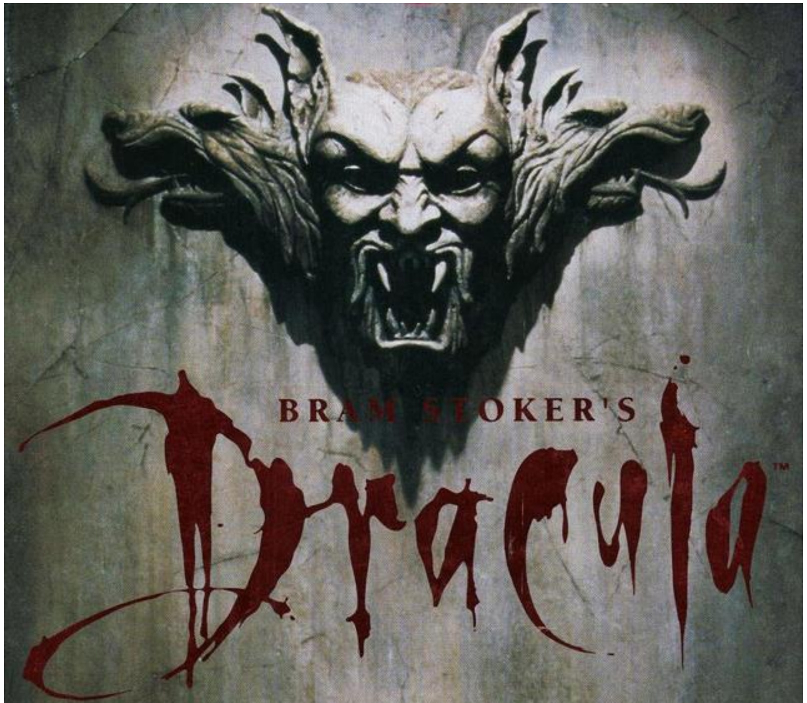
- ✧ Good project management is essential if software engineering projects are to be developed on schedule and within budget.
- ✧ Software management is distinct from other engineering management. Software is intangible. Projects may be novel or innovative with no body of experience to guide their management. Software processes are not as mature as traditional engineering processes.
- ✧ Risk management involves identifying and assessing project risks to establish the probability that they will occur and the consequences for the project if that risk does arise. You should make plans to avoid, manage or deal with likely risks if or when they arise.

Key points

- People management involves choosing the right people to work on a project and organizing the team and its working environment.
- People are motivated by interaction with other people, the recognition of management and their peers, and by being given opportunities for personal development.
- Software development groups should be fairly small and cohesive. The key factors that influence the effectiveness of a group are the people in that group, the way that it is organized and the communication between group members.
- Communications within a group are influenced by factors such as the status of group members, the size of the group, the gender composition of the group, personalities and available communication channels.

Personality types

Difficult people to work with: emotional vampires



- Not just annoying people.
- *Emotional Vampires* draw you in with illusion (e.g., talent).
- Then they drain you – of every last drop of emotional energy.
- They can’t see their reflection in the mirror…

BULLIES

- ✧ They are addicted to the raw thrill of seeing you squirm.
- ✧ LOOK FOR: Anger, threats, and yelling.
- ✧ DRAW YOU IN WITH: The illusion that they are powerful.
- ✧ DRAIN YOU BY: Making you afraid.
- ✧ THE ONES YOU SEE EVERY DAY: petty tyrant who runs the finance department.
- ✧ DEFENSIVE STRATEGY: Remember that the real battle with bullies is not in the dust of the playground, but in your own mind.

USED-CAR SALESMEN

- Their drug of choice is putting one over on you.
- LOOK FOR: People who swear they're telling the truth. Think about it, who but a liar would expect to be doubted?
- DRAW YOU IN WITH: Instant rapport, smiling sincerity, and the sweet prospect of something for nothing.
- DRAIN YOU BY: Lying, cheating, stealing, and perhaps getting you to engage in a bit of subterfuge yourself.
- THE ONES YOU SEE EVERY DAY: Anybody who asks you if you've thought about your future.
- DEFENSIVE STRATEGY: Always read the fine print, and remember, if a deal sounds too good to be true, it is.

Histrionic

- ✧ Histrionic means dramatic.
- ✧ With these vampires everything’s a show, and what you see is seldom what you get.
- ✧ They crave attention and approval and will do anything to get them, except making good on their promises.
- ✧ **They** live to put on a show. If you don't pay enough attention they'll, they'll just shrivel up and die. It will be your fault.

Histrionic

- LOOK FOR: the always nice syndrome. The victim syndrome. Histrionics have big emotions and not the tiniest clue of who they really are.
- DRAW YOU IN WITH: Charm, beauty, talent, or whatever else it takes to get noticed, except for substance.
- DRAIN YOU BY: Needing more maintenance than a classic car, but having almost nothing to give back.
- THE ONES YOU SEE EVERY DAY people whose emotions are a mile wide and an inch deep.
- DEFENSIVE STRATEGY: Histrionic lives are soap operas. Watch, but don't join the cast.

Passive-aggressive

- They long for approval so much that they hide any part of themselves that anyone would disapprove of, especially anger.
- LOOK FOR: Bright smiles, blatant denial
- DRAW YOU IN WITH: Being nice, friendly, cheerful, enthusiastic, brave, clean, and reverent.
- DRAIN YOU BY: Attacks that everyone else sees, but *they* never acknowledge, or by simply being too forgetful to give you what they promise. Or maybe they're sick with some vague and undiagnosible illness.

Passive-aggressive

- THE ONES YOU SEE EVERY DAY: People who never get angry themselves, but seem to make everybody else angry. Anyone who would *never think* of doing anything unacceptable.
- DEFENSIVE STRATEGY: First and foremost, never try to get Passive-Aggressives to admit to their own motivation; you will only get a headache.
  - Remember that they hunger for approval.
  - Tell them explicitly what it takes to please you and praise them profusely when they do it.
  - The strategy is simple and almost foolproof, but it is seldom employed because it's hard to praise somebody who gives you headaches. Hard as it is, it's far easier than the alternative.

Narcissistic

- **LEGENDS IN THEIR OWN MINDS** with talent like theirs, who needs performance?
  - LOOK FOR: Self-proclaimed genius, with membership in MENSA listed as greatest life achievement.
  - DRAW YOU IN WITH: Talent and potential.
  - DRAIN YOU BY: Blaming everybody else for their inability to realize that potential.
  - THE ONE YOU SEE EVERY DAY: the genius who can't hold down a job; arrogant people.
  - DEFENSIVE STRATEGY: Getting them to do the parts they don't like first, and paying attention to performance rather than talk.

Narcissistic checklist

- THIS PERSON HAS ACHIEVED MORE THAN MOST PEOPLE HIS OR HER AGE.
- THIS PERSON IS FIRMLY CONVINCED THAT HE OR SHE IS BETTER, SMARTER, OR MORE TALENTED THAN OTHER PEOPLE.
- THIS PERSON LOVES COMPETITION, BUT IS A POOR LOSER.
- THIS PERSON HAS FANTASIES OF DOING SOMETHING GREAT OR BEING FAMOUS, AND OFTEN EXPECTS TO BE TREATED AS IF THESE FANTASIES HAD ALREADY COME TRUE.
- THIS PERSON HAS VERY LITTLE INTEREST IN WHAT OTHER PEOPLE ARE THINKING OR FEELING, UNLESS HE OR SHE WANTS SOMETHING FROM THEM.
- TO THIS PERSON IT IS VERY IMPORTANT TO LIVE IN THE RIGHT PLACE AND ASSOCIATE WITH THE RIGHT PEOPLE.
- THIS PERSON TAKES ADVANTAGE OF OTHER PEOPLE TO ACHIEVE HIS OR HER OWN GOALS.

Obsessive-compulsive

- They are perfectionists who dot every *i*, cross every *t,* but can't seem to see the forest for the for the superfluous overabundance of trees.
- LOOK FOR: Overwork and emotional constipation.
- DRAW YOU IN WITH: Competence, hard work, and attention to detail.
- DRAIN YOU BY: Punishing everybody who's not as competent as they are.
- THE ONES YOU SEE EVERY DAY: Puritans, perfectionists, and control freaks.
- DEFENSIVE STRATEGY: Negotiate to deliver a product, then resist attempts to control the process

Paranoid Vampires

- They are visionaries who search hard for the Truth. They see things that others can't. The question is whether those things are really there.

  LOOK FOR: Absolute certainty about themselves and their beliefs, distrust of everything else.
- DRAW YOU IN WITH: Virtue, creativity, perceptiveness, protectiveness.
- DRAIN YOU BY: Jealousy, unshakable prejudices, constant cross-examination of every difference of opinion. Absolute belief is neither good nor evil, but it is a door through which either may enter.
- THE ONES YOU SEE EVERY DAY: conspiracy theorists, and people who post weird things on the Internet.
- DEFENSIVE STRATEGY: Know how to recognize a crazy idea when you hear one. Don't answer questions when they're asked a second time.

How to Protect Yourself

KNOW THEM, KNOW THEIR HISTORY, AND KNOW YOUR GOAL

- The big mistake you can make with vampires is assuming, without evidence, that though their record has been bad in the past, that they have learned their lesson, and will do better this time.

GET OUTSIDE VERIFICATION

- Vampires want you to listen to them alone. To control you, they'll try to isolate you from your usual sources of information. Always check out what they say with a trusted friend, especially when you'd rather not. Vampires can't operate in the light of day.

PAY ATTENTION TO ACTIONS, NOT WORDS

- What vampires say is often very different from what they do. To avoid being drained, always focus on what they do.

How to protect

CHOOSE YOUR WORDS AS CAREFULLY AS YOU PICK YOUR BATTLES

- With Emotional Vampires what you say, how you say it, and when you say it are all crucial to the outcomes you are likely to achieve.

IGNORE TANTRUMS

- When vampires don't get their way, they throw tantrums. They can explode into all sorts of emotional outbursts whose only purpose is to get you to give in. Don't.

KNOW YOUR OWN LIMITS

- Dealing with Emotional Vampires requires a lot of effort. They may be worth it, they may not. Only you can decide. Sometimes it's better to run away, or not get involved in the first place.