



DECSAI

Departamento de Ciencias de la Computación e I.A.

Universidad de Granada



Peopleware

Fernando Berzal, berzal@acm.org

Peopleware



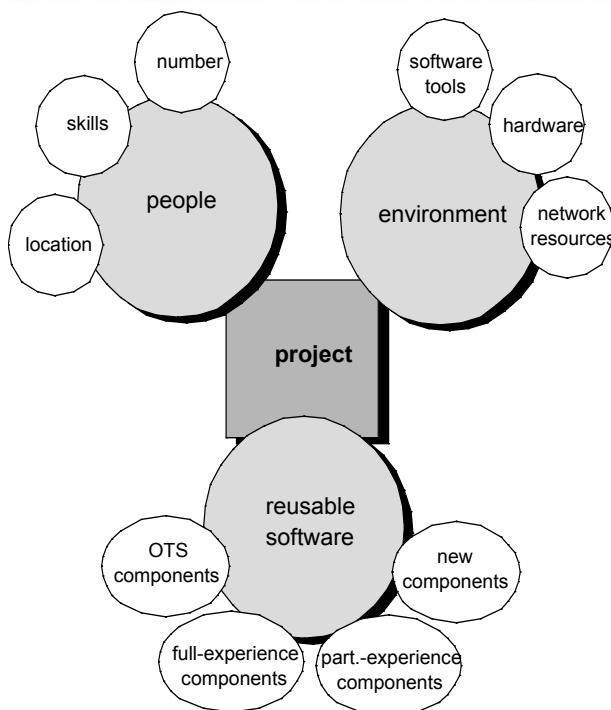
- El gestor como catalizador
- El factor humano
- Motivación
- Innovación
- El equipo del proyecto
- Selección de personal
- El entorno de trabajo
- Reuniones
- Evaluaciones de rendimiento
- Desarrollo profesional
- Cultura empresarial
- Código ético



El gestor como catalizador



Recursos



Roger S. Pressman:
"Software Engineering: A Practitioner's Approach", 7th edition, 2009



El gestor como catalizador



La ley de la variedad necesaria

[Law of Requisite Variety]

W. Ross Ashby

Para que un sistema sea estable, el número de estados de su mecanismo de control debe ser igual o mayor que el número de estados del sistema controlado.

- COROLARIO: Como los elementos más complejos de un proyecto son las personas que participan en él, las personas son las únicas con la complejidad suficiente para gestionar el proyecto (ni los procesos ni las herramientas son suficientes).



El gestor como catalizador



Niveles de autoridad

- Ordenar
- Vender
(el gestor toma la decisión y se la vende al resto del equipo)
- Consultar
(se ponderan las opiniones del equipo, pero el gestor decide)
- Acordar
(se alcanza un consenso)
- Aconsejar
(el gestor da su opinión, pero la decisión recae en el equipo)
- Preguntar
(el equipo decide y luego intenta convencer al gestor)
- Delegar
(el equipo decide [e informa al gestor]).



El gestor como catalizador



Principio de oscuridad [darkness principle]:

Los agentes de un sistema complejo no conocen el sistema completo (si lo hiciesen, su complejidad completa residiría en el agente).

Teorema de Conant-Ashby

Todo buen regulador de un sistema debe tener un modelo del sistema que controla.

- COROLARIO: La delegación de control es la mejor forma de mantener los proyectos bajo control (la herramienta de la que dispone el gestor para controlar un sistema complejo).



El gestor como catalizador



El gestor debe considerar
la delegación de autoridad como una **inversión**

Johanna Rothman & Esther Derby: Behind Closed Doors, 2005

- El ROI tardará en llegar. Mientras tanto, costará tiempo, energía, dinero y alguna que otra frustración.
- Tomar las riendas sin dejar que los miembros del equipo adquieran autonomía es como sacar dinero del banco antes de que dé interés

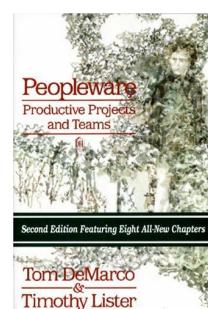


El factor humano



People under pressure don't work better;
they just work faster

-- Tom DeMarco



Quality is free,
but only to those who are willing to pay heavily for it.

-- Tom DeMarco

The manager's function is not to make people work,
but to make it possible for people to work.

-- Tom DeMarco



El factor humano



Las 4 P's:

- **Personas** (el elemento más importante para que un proyecto tenga éxito).
- **Producto** (el software que ha de construirse).
- **Proceso** (el conjunto de actividades y tareas que permiten realizar el trabajo).
- **Proyecto** (el trabajo necesario para que el producto sea una realidad).

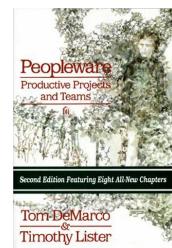
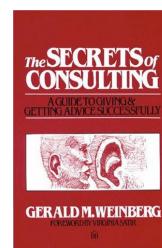


El factor humano



The Second Law of Consulting:
No matter how it looks at first,
it's always a people problem.

-- **Gerald M. Weinberg:**
"The Secrets of Consulting", 1985



La causa más frecuente de fracaso de un proyecto de desarrollo de software es "política": falta de comunicación, problemas de personal, desencanto con jefes o clientes, abandono de miembros clave del equipo...

Los problemas principales a los que se enfrenta un proyecto no son técnicos, sino "sociológicos".



El factor humano



"Stakeholders"

(personas con algún interés en el proyecto)

- Gestores senior / ejecutivos
(definen los objetivos de la empresa).
- Gestores técnicos / gestores del proyecto
(planifican, motivan, organizan y controlan).
- Desarrolladores / ingenieros
(proporcionan las habilidades técnicas necesarias)
- Clientes
(determinan los requisitos del sistema)
- Usuarios
(interactúan con el sistema).



El factor humano



"Stakeholders"

dentro de la organización

- Recursos humanos (contrataciones, despidos, becas, evaluaciones de rendimiento...).
- Finanzas (facturas, pagos, nóminas).
- Compras (adquisición de material).
- Departamento legal
(contratos, licencias, propiedad intelectual).
- Marketing (investigaciones de mercado, notas de prensa, publicidad, eventos corporativos).
- Ventas (tendencias, necesidades de los clientes...)
- Servicio técnico (problemas de los clientes...)



El factor humano



“Stakeholders”

frente a la organización

- Clientes
- Consultores externos
- Proveedores
- Gobiernos (nacional, regional, local)
- Organizaciones de estándares (ISO, ANSI...)
- Consorcios industriales
- Asociaciones profesionales (ACM, IEEE...)
- Contactos académicos/profesionales
- Contactos locales



El factor humano



- ¿Cómo dirigir?
- ¿Cómo liderar?
- ¿Cómo motivar?
- ¿Cómo organizar?
- ¿Cómo innovar?
- ...



“Management is getting people to do what needs to be done. Leadership is getting people to want to do what needs to be done.”

— Warren Bennis, USC

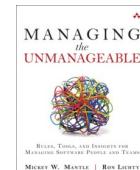
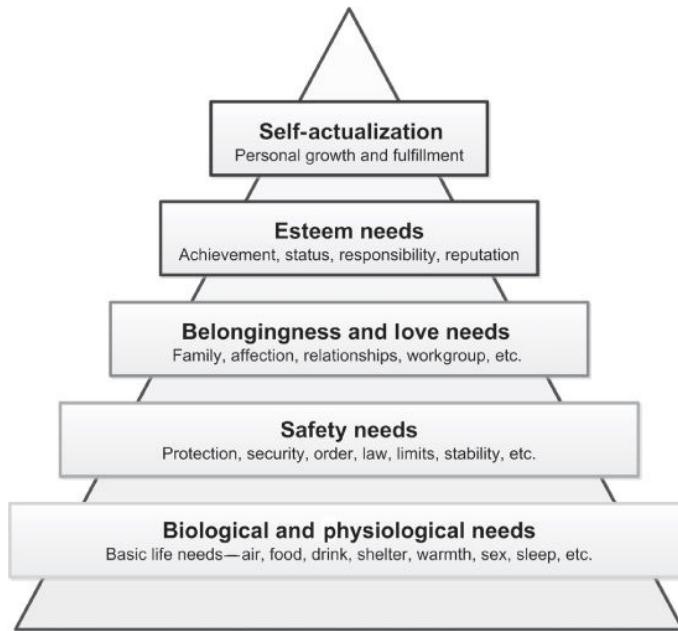


Motivación



Pirámide de necesidades de Maslow

Abraham Maslow: Motivation and Personality, 1954

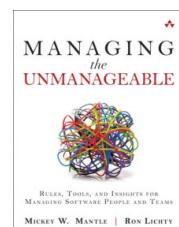
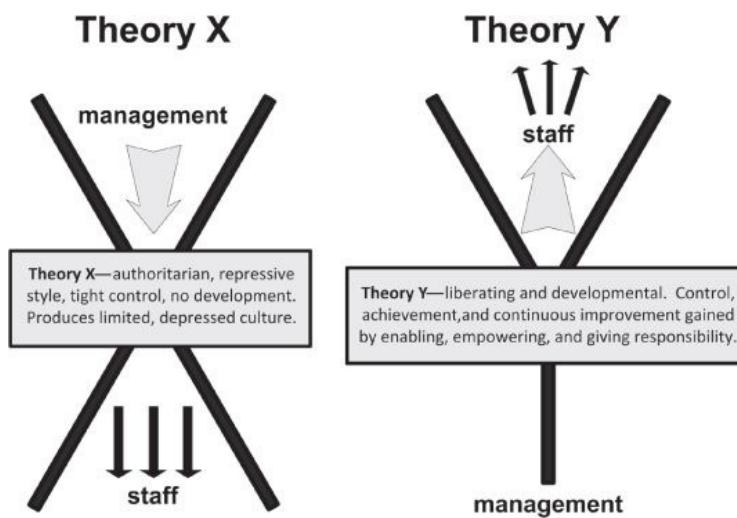


Motivación



Teoría X-Y

Douglas McGregor: The Human Side of Enterprise, 1960



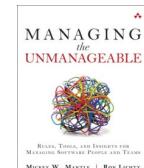
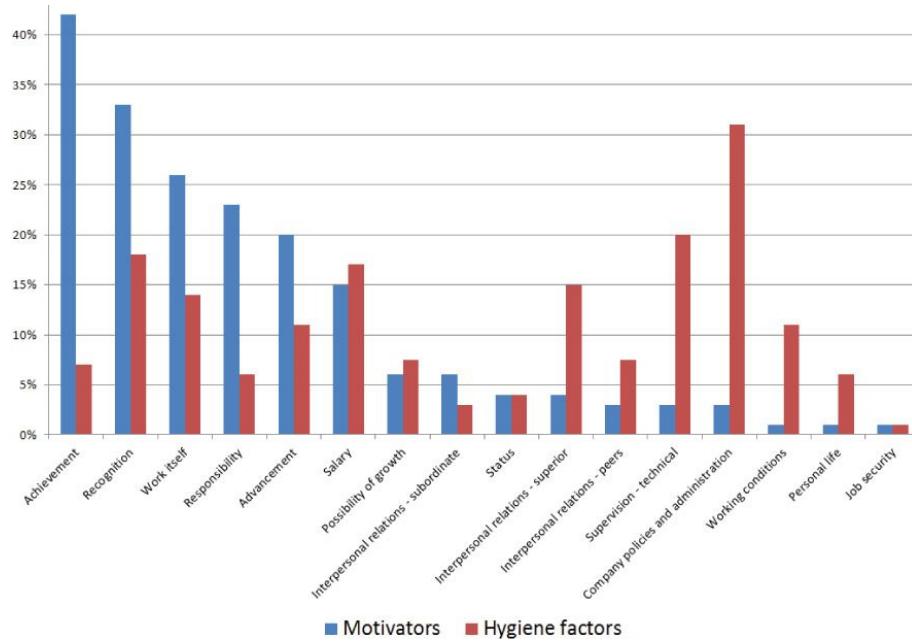
Motivación extrínseca vs. motivación intrínseca

Motivación



Motivadores y factores de higiene

Frederick Herzberg et al.: The Motivation to Work, 1959

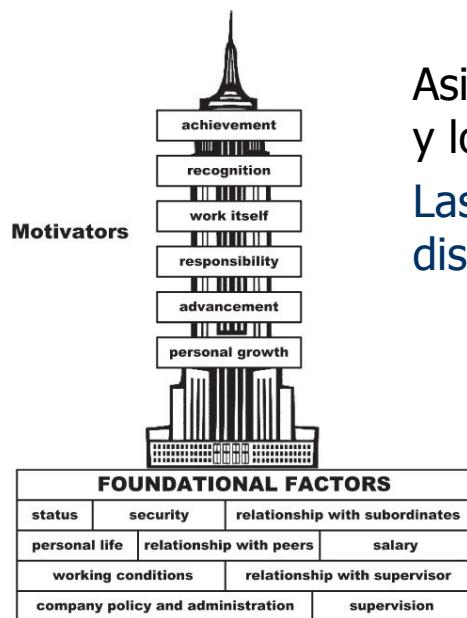


Motivación

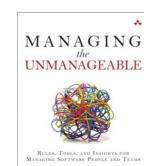


Motivadores y factores de higiene

Frederick Herzberg et al.: The Motivation to Work, 1959



Asimetría entre lo que motiva
y lo que desmotiva:
Las causas de insatisfacción son
distintas de los factores que motivan.

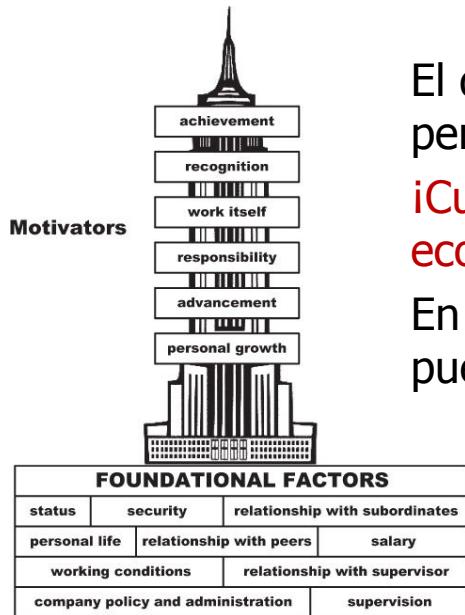


Motivación

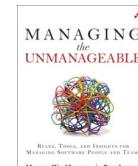


Motivadores y factores de higiene

Frederick Herzberg et al.: The Motivation to Work, 1959



El dinero no es un motivador, pero puede ser un desmotivador:
iCuidado con asociar recompensas económicas al logro de objetivos!
En entornos creativos, pueden resultar contraproducentes...



Motivación



"Using money as a motivator is like playing with dynamite because money is a VERY effective motivator. Monetary rewards motivate people to do EXACTLY what is being rewarded – not necessarily what the organization intended to reward, but EXACTLY what is being measured to generate the reward. Therefore monetary motivators have a long track record of generating unintended consequences. If there is any apparent competition for the money, money motivates people to get as much as they can for themselves. Thus monetary motivators have a track record of suppressing collaboration. Finally, bonuses for performance rapidly come to be an expected part of the landscape, replacing passion and dedication as motivators. These are things you probably cannot change about using money as a motivator."

-- Mary Poppendieck, Lean Software Development group, 2008

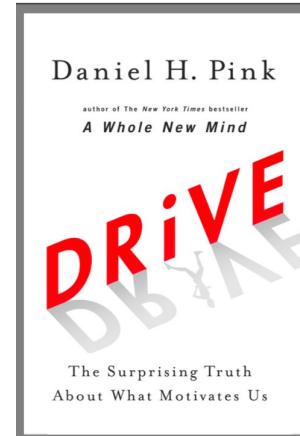


Motivación



Las personas con motivaciones intrínsecas (autonomía o autodeterminación, propósito [al servicio de un objetivo mayor], sentido del deber cumplido, realimentación positiva...) son más productivas que las que únicamente buscan obtener una recompensa.

Cuando se usan recompensas para motivar es cuando más desmotivan...



Daniel Pink: "Drive: The Surprising Truth about What Motivates Us", 2009

Motivación



"The more prominent salary, perks, and benefits are in someone's work life, the more they can inhibit creativity and unravel performance. . . .

Effective organizations compensate people in amounts and in ways that allow individuals to mostly forget about compensation and instead focus on the work itself. Here are three key techniques:

- Ensure internal and external fairness . . .
- Pay [a little] more than average . . .
- If you use performance metrics, make them wide-ranging, relevant, and hard to game."

— **Daniel H. Pink**

Daniel Pink: "Drive: The Surprising Truth about What Motivates Us", 2009



Motivación



4 de cada 5 empleados (el 79%) prefieren otros beneficios antes que un aumento de sueldo:
Seguros médicos, vacaciones pagadas, planes de pensiones, horarios flexibles, comidas gratis, desarrollo profesional, gimnasios, stock options, guarderías, transporte...



Glassdoor Employment Confidence Survey (Q3'2015)
<http://www.glassdoor.com/blog/ecs-q3-2015>



Innovación

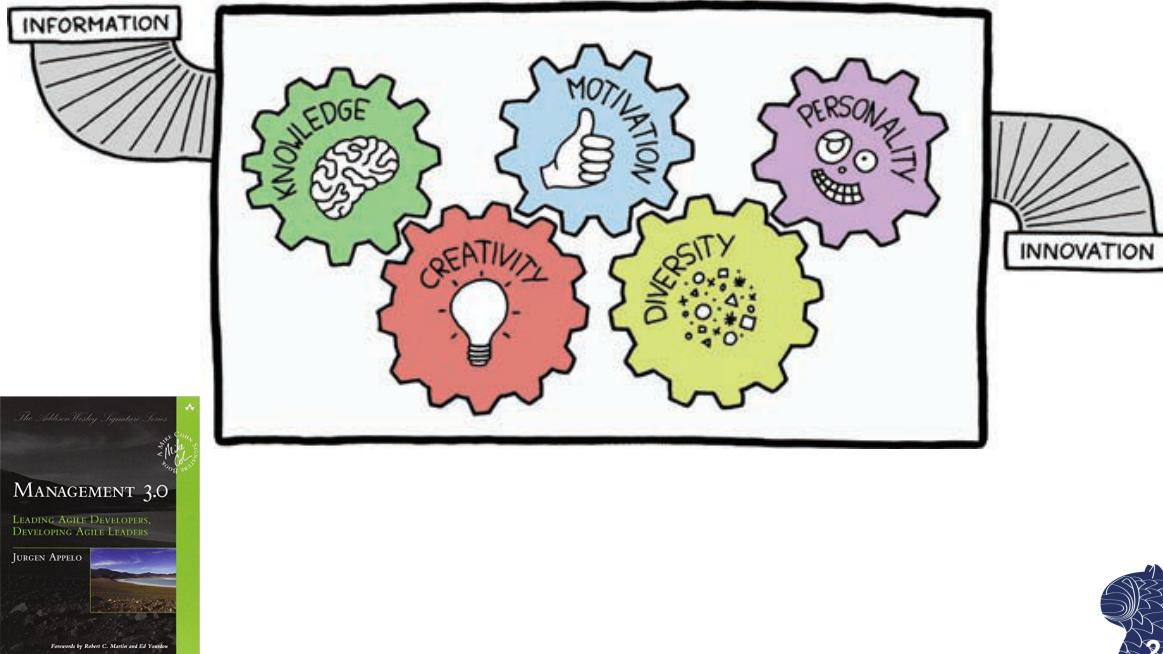


En cualquier entorno competitivo,
la innovación es clave para sobrevivir:

- La innovación es típicamente un fenómeno ascendente (no se puede imponer desde arriba en la jerarquía), p.ej. encargarle a alguien que invente algo.
- La innovación no es un resultado planeado, sino un resultado emergente.
- Sí que se pueden tomar medidas para facilitarla...



Innovación



Innovación



Conocimiento

Fuel necesario para la innovación

- La experiencia y pericia de alguien no suele ser el mejor predictor de su rendimiento.
- Lo que suele marcar la diferencia es su conectividad dentro de la organización: “red de conocimiento”.
Rob Cross et al.: **The Hidden Power of Social Networks**, 2004



Innovación



Creatividad

La variable crucial para convertir conocimiento en valor

El 80% del trabajo de software es intelectual.

Una parte representativa de él (16%) es creativo.

Muy poco es rutinario [clerical]



Robert L. Glass:

Facts and Fallacies of Software Engineering

Addison-Wesley, 2003. ISBN 0321117425

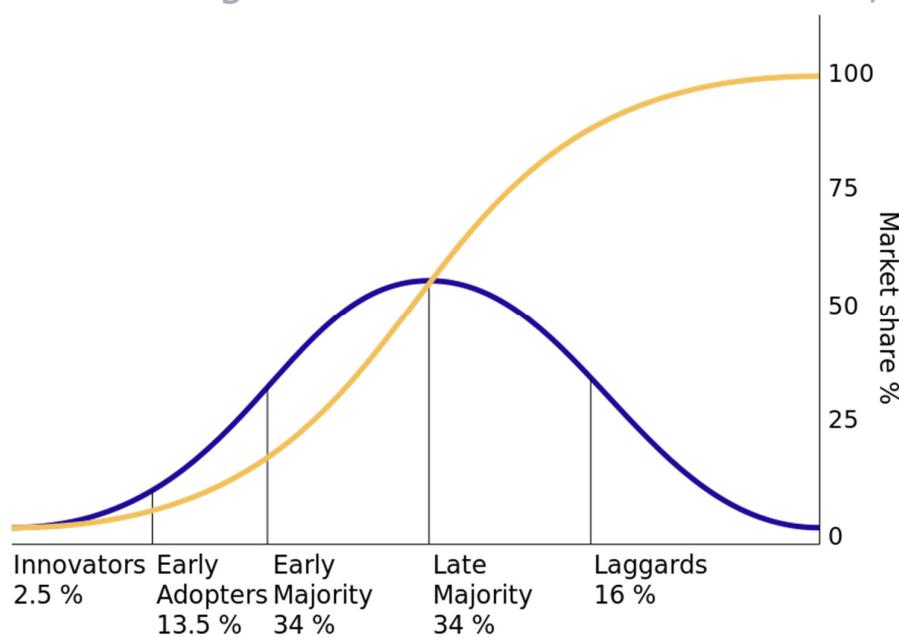


Innovación



La curva de la innovación

Everett Rogers: **Diffusion of Innovations**, 1962



Innovación



Riesgo & Innovación

- Para fomentar la innovación,
se debe fomentar la toma de riesgos.
- Para fomentar la toma de riesgos,
se deben tolerar los fracasos.
- Para tolerar los fracasos y fomentar la innovación,
se debería premiar la dedicación y la profesionalidad,
no el éxito...

Los errores no deben verse como algo que evitar por completo, sino como un mecanismo de aprendizaje

-- Gerald M. Weinberg:
Quality Software Management, 1992



Innovación



Política de “introducción constructiva de desorden” [“constructive reintroduction of small amounts of disorder”]

“The opposite approach [to a defined process] would be one in which every new undertaking is run as a pilot project. To the extent that there was a standard way to carry out the work, that would be the only way you *weren’t* allowed to carry it out. The standard would be for at least one part of the effort to be run in a nonstandard way.”

-- Tom DeMarco & Tim Lister:
Peopleware: Productive Projects and Teams
Dorset House, 2nd edition, 1999.



El equipo del proyecto



Factores que han de considerarse para decidir cómo organizar el equipo:

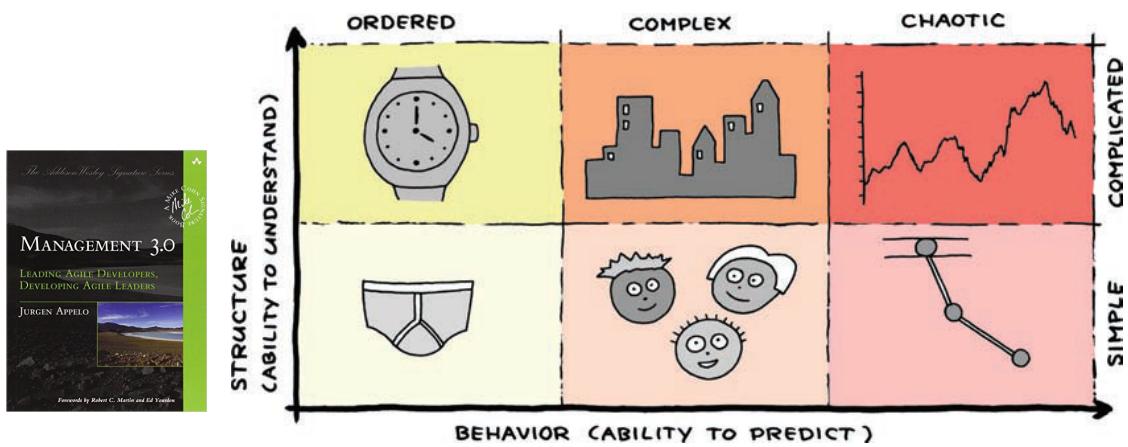
- Dificultad y complejidad del problema.
- Tamaño del proyecto.
- Esperanza de vida del equipo (tiempo durante el que los integrantes del equipo trabajarán juntos).
- Grado en que el problema se puede modularizar (descomponer en tareas independientes).
- Requisitos de calidad y fiabilidad.
- Rígidez de la fecha de entrega.



El equipo del proyecto



El equipo de un proyecto es un **sistema adaptativo complejo** porque está formado por múltiples individuos que interactúan, tiene una frontera definida y tiene la capacidad de cambiar y aprender de la experiencia.



El equipo del proyecto



Paradigmas organizativos

- **Cerrado** (jerarquía tradicional).
- **Aleatorio** (deja libertad absoluta, depende de la iniciativa individual de los miembros del equipo).
- **Abierto** (incorpora alguno de los controles del paradigma cerrado, intentando conservar la innovación que permite el paradigma aleatorio).
- **Síncrono** (confía en la división natural del problema, para que los miembros del equipo trabajen de forma independiente, con menor comunicación entre ellos).

Larry Constantine: "Work Organization: Paradigms for Project Management and Organization", CACM, vol. 36, no. 10, October 1993, pp. 34–43.



El equipo del proyecto



Algo que evitar: **TOXICIDAD**

(atmósfera de trabajo en la que los miembros del equipo desperdician energía y pierden de vista los objetivos del trabajo que hay que realizar).

- La frustración crea fricciones.
- La falta de coordinación o un modelo de proceso mal elegido bloquea el proyecto.
- La definición ambigua de responsabilidades y roles genera acusaciones cruzadas.
- El fracaso genera una pérdida de confianza y mina la moral de los miembros del equipo.



El equipo del proyecto



"... a good manager does not intimidate, condescend, demean, act arrogant, withhold praise, slam doors, pound tables, swear, behave rudely, belittle people in front of others, give mostly negative feedback, yell at people, tell lies or "half-truths," act above the rules, enjoy making people sweat, act superior to or smarter than everyone else, act sexist, act bigoted, withhold critical information, use inappropriate humor, blow up in meetings, steal credit or the spotlight from others, block career moves, show favoritism, humiliate or embarrass others, overuse sarcasm, deliberately ignore or isolate people, set impossible goals or deadlines, let others take blame for their mistakes, undermine authority, show lack of caring for people, betray confidence, gossip or spread rumors, act as if others are stupid, use fear as a motivator, show revenge, interrupt constantly, fail to listen, demand perfection, or break promises. And these are, of course, just a few examples of things you should not do."

-- Jurgen Appelo: **Management 3.0**



El equipo del proyecto



Coordinación y comunicación

- Medios impersonales formales: documentos y productos del trabajo (incluido el código fuente), informes técnicos, plan del proyecto, solicitudes de cambios, informes de errores...
- Medios interpersonales formales: evaluaciones de rendimiento, reuniones de seguimiento, inspecciones de diseño y de código...
- Medios interpersonales informales: reuniones de grupo, conversaciones "de café", e-mails...



El equipo del proyecto



Coordinación y comunicación

Wisdom is the reward you get for a lifetime of listening when you'd rather have been talking.

— ARISTOTLE

The single biggest problem in communication is the illusion that it has taken place.

— GEORGE BERNARD SHAW



El equipo del proyecto



Coordinación y comunicación

MBWA [Management by Walking Around]

e.g. Bob Hewlett & David Packard, HP

- Tan importante como las reuniones personales con los miembros del equipo.
- Muestra interés en las personas del equipo y en lo que hacen, además de aprender realmente lo que hacen.
- No debe hacerse de forma “predecible”.
- Las discusiones y críticas, mejor a puerta cerrada.



El equipo del proyecto



Problemas de coordinación y comunicación

- **Versionitis:** Dos programadores trabajan en módulos diseñados para encajar correctamente, pero divergen.



El equipo del proyecto



El gestor debe crear los incentivos necesarios para...

- Minimizar la rotación de personal [turnover].
- Garantizar la calidad del trabajo realizado.
- Evitar interferencias externas que hagan disminuir la productividad del equipo.



El equipo del proyecto



Para evitar que disminuya la productividad:

- La burocracia se ha de reducir al mínimo.
- Las reuniones improductivas han de eliminarse.
- La adherencia dogmática a procesos y reglas no debe ser impuesta.
- El equipo debe poder auto-organizarse y disfrutar de cierto grado de autonomía.



El equipo del proyecto



Tamaño ideal del equipo

Conforme el equipo aumenta de tamaño, la comunicación entre los miembros del equipo empeora: las interacciones, más fragmentadas, aumentan el número de falsas suposiciones y errores que podrían evitarse en equipos más pequeños.

Los equipos grandes requieren una planificación más rigurosa, medios de comunicación formalizados, una documentación más exhaustiva y un sistema integrado de pruebas.



El equipo del proyecto



Tamaño ideal del equipo

"Small teams are more productive than large teams. . . . A 5- to 7-person team will complete an equivalently sized project in the shortest amount of time."

— **Mike Cohn:** "Succeeding with Agile", p. 181.

Putnam, en QSM, estudió 491 proyectos y descubrió que, cuanto más pequeños eran los equipos, más productivos eran los miembros individuales del equipo, aunque las diferencias eran pequeñas en equipos de 2 a 7 personas.



El equipo del proyecto



Tamaño ideal del equipo

$B=f(P,E)$ Ecuación de Lewin: el comportamiento depende de la personalidad y del entorno.

$C=f'(P,E)$ Variante: la comunicación depende de la personalidad y del entorno

$S=f''(\{P\},E)$ El tamaño del equipo (básicamente, un problema de comunicación), depende de las personalidades de sus integrantes y del entorno en el que trabajan:
No existe un valor óptimo para S :-(



El equipo del proyecto



Ley de Conway: Las organizaciones que diseñan sistemas... están limitadas a producir diseños que son copias de sus estructuras de comunicación.

"If you have four groups working on a compiler, you'll get a 4-pass compiler."

-- Eric S. Raymond: The New Hacker's Dictionary

Por otro lado, es mejor mantener los equipos que funcionan tanto como sea posible a lo largo de distintos proyectos, ya que lleva tiempo que se formen las estructuras de comunicación y las reglas que hacen que el equipo "cuaje" y sea más que la suma de sus partes.

44

El equipo del proyecto



Ley de Lunde

"Beware of Lunde's Law: Any time the senior developers on a project are replaced by a new team, the new team will eventually find a compelling reason for a total rewrite. And that new team is wrong." — Luke Hohmann

Ley de Brooks

"Adding manpower to a late software project makes it later"

— Frederick P. Brooks: The Mythical Man-Month, 1975



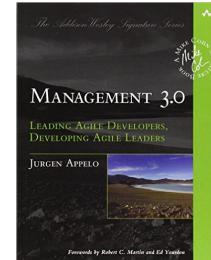
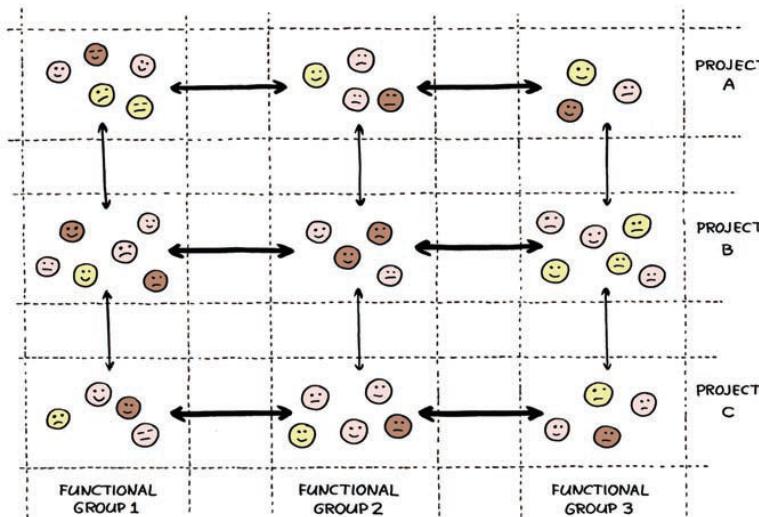
45

El equipo del proyecto



Organización de los equipos

Equipos funcionales vs. Equipos multidisciplinares

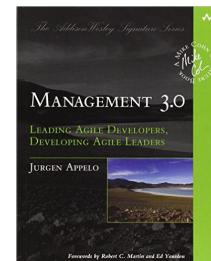
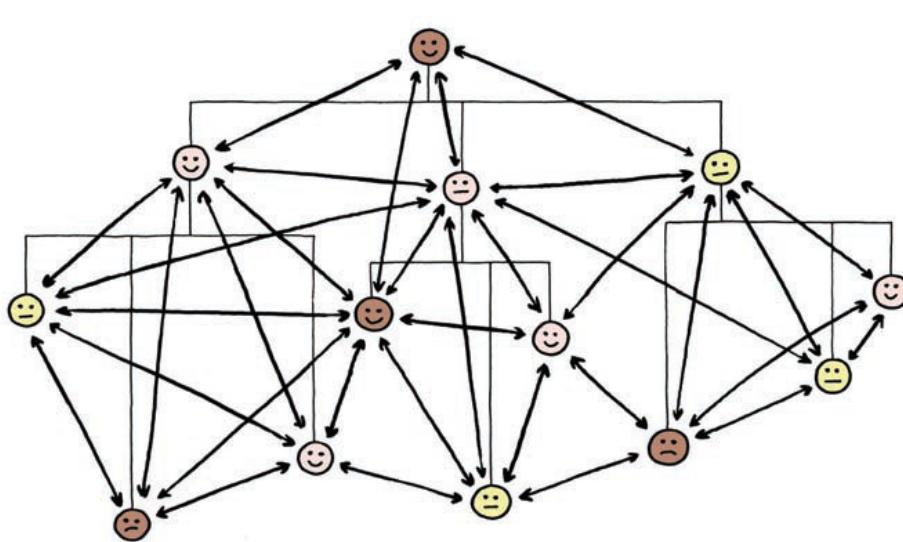


El equipo del proyecto



Organización de los equipos

Equipos funcionales vs. Equipos multidisciplinares



El equipo del proyecto



Organización de los equipos

¿Cuándo añadir un nuevo nivel jerárquico?

Cuando quede claro que ese nivel va a resolver problemas que no pueden resolver ni los niveles inferiores ni los niveles superiores

¿Cuántos “gestores” tiene que haber?

- 1 por equipo

[Louis Testa: Growing Software, 2009]

- 1 por cada 100 empleados

[Craig Larman & Bas Vodde: Scaling Lean & Agile Development, 2009]

Dependerá de

cómo haya crecido la estructura de la organización...



El equipo del proyecto



Equipos distribuidos

“Communication is a factor of the inverse square of the distance domestically, and the inverse cube of the distance internationally”

— David C. Evans, E&S

There's no substitute for face-to-face communication, particularly at pivotal points in the project.

— Ade Miller, Microsoft patterns & practices group

The more distance between teammates, the more you have to formalize communication and make it explicit.

— Ted Young, GuideWirE



El equipo del proyecto



I've learned that people will forget what you said, people will forget what you did, but people will never forget how you made them feel.

-- Maya Angelou

Don't take chances on team jell if you don't have to:
Seek out and use preformed teams.

Keep good teams together (when they're willing) to help your successors avoid problems of slow-jelling or non-jelling teams.

Think of a jelled team—ready and willing to take on a new effort—as one of the project deliverables.

— Tom DeMarco: "The Deadline"



Selección de personal



A's hire A's. B's hire C's.

— Steve Jobs

Exceptional engineers are more likely than non-exceptional engineers to maintain a 'big picture,' have a bias for action, be driven by a sense of mission, exhibit and articulate strong convictions, play a pro-active role with management, and help other engineers.

— Richard Turley & James Bieman: "Competencies of Exceptional and Non-Exceptional Software Engineers", Colorado State University, 1993.



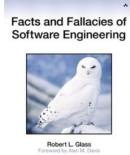
Selección de personal



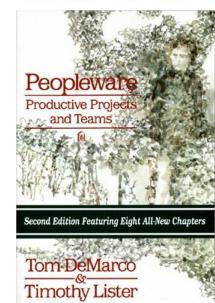
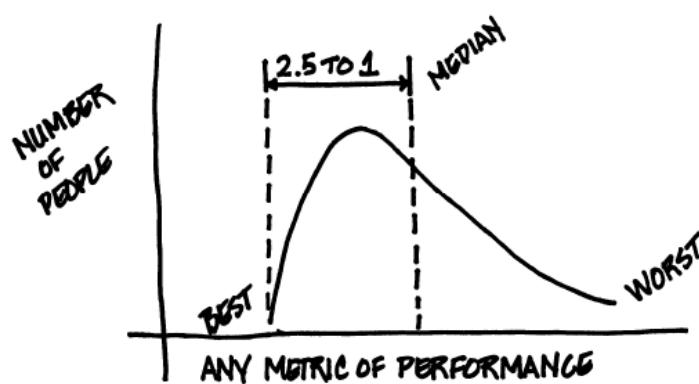
Existen diferencias de 10 a 1 e incluso de 20 a 1 en el rendimiento de diferentes personas y equipos.

En los sueldos no existen esas diferencias, por lo que seleccionar adecuadamente al personal de un proyecto resulta ser una inversión de lo más rentable...

The best programmers are up to 28 times better than the worst programmers, according to "individual differences" research. Given that their pay is never commensurate, they are the biggest bargains in the software field. — **Robert L. Glass**



Selección de personal



H. Sackman, W. J. Erikson & E. E. Grant, "Exploratory Experimental Studies Comparing Online and Offline Programming Performance," Communications of the ACM, January **1968**.

Gerald M. Weinberg & Edward L. Schulman: "Goals and Performance in Computer Programming," Human Factors 16(1):70-77, **1974**.

Barry Boehm: "Understanding and Controlling Software Costs," IEEE Transactions on Software Engineering, October **1988**.

Barry Boehm et al.: Software Cost Estimation with Cocomo II, **2000**.

Selección de personal



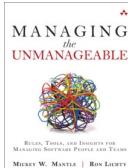
Descripción del puesto de trabajo

TITLE: Programmer 3
DEPARTMENT: Client Programming
REPORTS TO: Director, Client Programming
STATUS: Full-time, exempt
LOCATION: San Francisco, CA

POSITION SUMMARY: Entry-level position. Responsible for code and/or asset management, conversion, verification, and maintenance. Responsible for writing well-defined portions of source code adhering to established standards of quality for documentation and coding. Works well in a group, and follows direction from manager and senior team members. Expected to work under direct supervision, communicating issues and problems that arise.

JOB REQUIREMENTS

- Four-year college degree in computer science or equivalent experience.
- Knowledge of Windows, Mac, or Linux/UNIX with more than one platform preferable.
- Knowledge of C/C++ and debugging techniques.
- Basic knowledge of good coding practices and fundamental computer science principles.
- Aware of and interested in Internet technologies, communication protocols, and techniques.
- Aware of and interested in database methodologies and database systems.
- Ability to work in a team and take direction well.
- Self-motivated and responds to supervision. Asks relevant questions.
- Enthusiastic about company and programming company products.
- Can work with supervisor to plan tasks and estimate their completion.
- Can adapt to changing conditions.



Selección de personal



Descripción del puesto de trabajo

PRINCIPAL PROGRAMMER, .NET

San Francisco/Oakland/ Berkeley ← Specific location
(Note: significant telecommuting opportunity if desired) ← Telecommuting?
Competitive salary, benefits, and options ← Not equity only

Forensic Logic, Inc. (www.forensiclogic.com), is an early-stage, growth-oriented company looking for a highly productive senior developer with the ability to lead a team and set its technical direction based on tons of experience designing, coding, and scaling .Net and SQL Server high-volume Web and analytics applications.

Forensic Logic develops Web-based applications that provide law enforcement agencies with tools that facilitate increased officer safety, early detection of crime trends, and interagency search capabilities. The successful candidate will have a unique opportunity to work with massive data sets, both structured and unstructured, and extensive association, geospatial, timeline, and pattern analysis and visualization, and application of matching and ranking algorithms for solving crimes. The position will provide a growth opportunity to the right individual who will be part of a great team of talented and motivated coworkers.

Forensic Logic's culture values respect, teamwork, and collaboration in achieving leading-edge functionality balanced with high usability.

JOB DESCRIPTION

- Provide technical team leadership, direction, and mentoring to the small, existing remote programming team.
- Form the nucleus of a second Bay Area development team.

- Apply incisive design and exceptional coding skill to knocking features off the products' extensive and growing features list.
- Lead periodic rapid refactorings that keep the application code fresh, flexible, and reusable.
- Help define team development and engineering best practices.
- Lead the team's implementation of best practices.

REQUIRED SKILLS

- Strong Web application architecture and design skills
- Fast, clean, efficient code implementation
- In-depth knowledge of Microsoft .Net and SQL Server
- 3+ years' experience designing, developing, and scaling high-volume Web applications on .Net and SQL Server platforms
- Leadership and mentorship of other developers, junior and senior alike
- Team orientation; ability to participate in lively engineering debate, making a strong case for well-considered opinions, while listening to, appreciating, and critiquing the opinions of peers
- Ability to analyze and improve the scalability and performance of high-volume, information-rich Web applications
- Strong verbal and written communication skills
- Strong customer empathy and customer experience sensitivity
- Must be highly self-motivated, ambitious, flexible, self-sufficient, and high-energy
- 8+ years' programming experience

EXPERIENCE WITH ANY OF THE FOLLOWING A PLUS:

- Algorithmic design and implementation; reasoning through algorithmic trade-offs
- Search/information retrieval
- Analytics, data warehousing, and business intelligence
- Information visualization
- Web services

Some (but not extensive) travel will be required.

Located adjacent to BART in the heart of the San Francisco Bay Area.

www.ForensicLogic.com

Sent résumés to:
Ron Lichty
VP, Products and Engineering
Forensic Logic, Inc.
RLichty@ForensicLogic.com

No phone calls

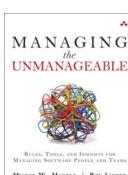
Principals only

The skills you consider a bonus

Travel?

Location

Contact information



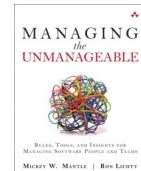
Selección de personal



Evaluación de currículos

Beforehand:

- List what you're looking for, in priority order
- Get clear on what your minimum requirements are
- List words and phrases that might be clues; add to your list as you read resumes



While reading:

- Highlight words that seem to identify the skills, tools, qualities you're seeking Mark gaps in employment history, to ask about
- Watch for job-hopping, if it's a concern
- Circle spelling errors, bad grammar, and sloppy formatting
- Jot questions the narrative raises: Are you clear what part the candidate played in the accomplishment listed? Are you clear what the effort accomplished? What design, tools, technologies, languages did the candidate use? How does the candidate grow their skillset through their career? Where and how did they learn their skills? Does the resume give any clue into whether they do stuff on their own?

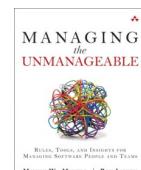


Selección de personal



Preselección de candidatos

Elaboración de un ranking de candidatos
(p.ej. evaluación subjetiva de 0 a 5
para cada criterio que se considere relevante)



Screening spreadsheet (for each interviewer/interviewee)

- Total Experience / Years of Experience
- Relevant Experience
- Management Experience (>5 programmers)
- Overall Technical Strength
- Specific Technical Skills
- ...
- Marketing/Product Focus
- Local/Relocation
- Personal Chemistry
- Comments



Selección de personal



Comprobación de referencias

Recopilación de información sobre el candidato

Candidato
Referencia
Capacidad
Puesto/responsabilidades del candidato
Evaluación de su rendimiento en el trabajo
Evaluación de sus conocimientos técnicos
Fortalezas
Habilidades interpersonales
¿Alguna crítica?
¿Volvería a contratarle / trabajar con él?
Comentarios



Selección de personal



Comprobación de referencias

[“back/hidden-channel” references]

“Never be satisfied talking only with the references your candidate supplies. If they’re a friend of the candidate, they often won’t mention the candidate’s faults—and everyone has them. Find an independent source—someone you know who has worked with the candidate as a peer as well as someone who managed or worked for them.”

— Dave Curbow, User Experience Architect, Cisco

p.ej. LinkedIn



Selección de personal



Entrevistas

I never hire candidates - for any position - from whom I've been unable to learn something during our interview. -- Ron Lichty

I have always found that getting candidates to talk about a project they have done in detail brings me the best info: how well they communicate, what roles they actually had, do they have a big picture about what they did, do they really understand the technical details. . . .

— Mark Himelstein



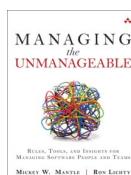
Selección de personal



Entrevistas

Principal Programmer Interview Summary	Bill Smith	Cathy Liu	Arnold Lai	Lucy Miller	Andy Jones
Received résumé on					
Phone screen on					
First interview round on					
On time, early, or late?					
Second interview round on					
On time, early, or late?					
Bachelor's Degree (optional)					
Minimum 8 years programming experience					
Wrote first program ever in (year, language)					
Wrote first professional program in					
Experience with what languages					
Experience with what databases					
Minimum 3 years .Net programming experience					
Wrote first .Net program in					
Most recently wrote for .Net in					
Minimum 3 years SQL Server programming experience					
Wrote first SQL Server program in					
Most recently wrote for SQL Server v. (???) in (year)					
Web application architecture and design skills?					
Ability to analyze & improve scalability and performance					
Experience scaling high-volume, information-rich Web apps					
Fast, clean, efficient coder?					
Refactoring skills					
Has defined development and engineering best practices					
Experience leading and mentoring other developers					

Communicates designs effectively				
Listens				
Critiques others' designs				
Writing skills				
Customer Experience empathy/ awareness/design sense				
Intangible qualities				
Energy				
Flexibility				
Self-direction				
Smart				
Articulate				
Passionate				
Fit in with team				
Overall desire to work at our company				
Experience w/algorithmic design, coding, trade-offs				
Search/information retrieval				
Analytics, data warehousing, and business intelligence				
Information visualization				
Web services				
Sent us a follow-up thank you?				



Selección de personal



Entrevistas

"On the hiring side, we found that brainteasers are a complete waste of time. How many golf balls can you fit into an airplane? How many gas stations in Manhattan? A complete waste of time. They don't predict anything. They serve primarily to make the interviewer feel smart."

Instead, what works well are structured behavioral interviews, where you have a consistent rubric for how you assess people, rather than having each interviewer just make stuff up."

-- **Laszlo Bock**, VP Operations, Google

<http://www.nytimes.com/2013/06/20/business/in-head-hunting-big-data-may-not-be-such-a-big-deal.html>



Selección de personal



Tras las entrevistas

Debriefing your interviewers, as a team, is critical: Not only is it an opportunity for you to understand the team's perspective, but for them, observing how others can perceive different aspects of the candidate can help each team member improve their interviewing skills.

— Phac Le Tuan



Selección de personal



El perfil ideal:

Especialistas generalizadores

[generalizing specialists, a.k.a. T-shaped people]

- Una o más especialidades técnicas.
- Conocimiento general de desarrollo de software.
- Conocimiento general del negocio/empresa.
- Proactivos en la adquisición de nuevas destrezas (tanto en sus especialidades como en otras áreas, técnicas y no técnicas).

Hacen un trabajo muy bien
y otros trabajos de forma adecuada.



Selección de personal



Atributos clave en la selección de candidatos

- El candidato tiene algo que enseñar al equipo (su presencia aportará cosas nuevas).
- Al candidato le encanta programar / la tecnología (su entusiasmo le permitirá avanzar).
- El candidato es agradable en el trato (soporta bien las críticas constructivas, sabe cuándo pedir ayuda, dispuesto a aprender nuevas técnicas y formas de trabajar...)



Selección de personal



Atributos clave en la selección de candidatos

- El candidato se muestra orgulloso de su trabajo (atención al detalle, énfasis en la legibilidad y reutilización de su código..)
- El candidato tiene carácter “emprendedor” (proactivo a la hora de resolver problemas, piensa cómo mejorar todos los aspectos de la empresa y del producto).



Selección de personal



¿Falta algo en la lista anterior?

¿Conocimientos técnicos?

¿Diseño de algoritmos?

¿Últimas modas en programación?

"Years ago, we did a study to determine whether anyone at Google is particularly good at hiring. We looked at tens of thousands of interviews, and everyone who had done the interviews and what they scored the candidate, and how that person ultimately performed in their job. We found zero relationship. It's a complete random mess, except for one guy who was highly predictive because he only interviewed people for a very specialized area, where he happened to be the world's leading expert."

-- **Laszlo Bock**, VP Operations, Google

<http://www.nytimes.com/2013/06/20/business/in-head-hunting-big-data-may-not-be-such-a-big-deal.html>



El entorno de trabajo



La ecuación de Lewin

Kurt Lewin: Principles of Topological Psychology, 1936

$$B = f(P, E)$$

B: Comportamiento

P: Persona

E: Entorno

- Las personas adaptan su comportamiento al entorno en el que se encuentran.



El entorno de trabajo



"A programmer is most productive with a quiet private office, a great computer, unlimited beverages, an ambient temperature between 68 and 72 degrees (F), no glare on the screen, a chair that's so comfortable you don't feel it, an administrator that brings them their mail and orders manuals and books, a system administrator who makes the Internet as available as oxygen, a tester to find the bugs they just can't see, a graphic designer to make their screens beautiful, a team of marketing people to make the masses want their products, a team of sales people to make sure the masses can get these products, some patient tech support saints who help customers get the product working and help the programmers understand what problems are generating the tech support calls, and about a dozen other support and administrative functions which, in a typical company, add up to about 80% of the payroll."

— Joel Spolsky



El entorno de trabajo



Why does the effect of pressure on programmers max out after only 6% productivity gain?

My answer: People under pressure don't think faster.

— **Tom DeMarco:** "The Deadline"



El entorno de trabajo



Espacios abiertos vs. Despachos individuales

Ventajas de los espacios abiertos:

- Flexibilidad.
- Facilidad de comunicación.
- Reducción de costes.

Desventajas de los espacios abiertos:

- Distracciones & interrupciones
- Recursos compartidos
(climatización, ruido, iluminación...)
- Sin autoridad central: "tragedia de los comunes"



El entorno de trabajo



Espacios abiertos vs. Despachos individuales

Google

Microsoft

facebook.

IBM

pwc

McKinsey&Company



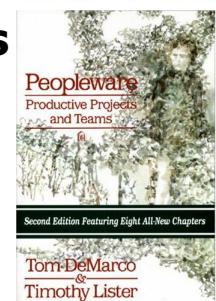
El entorno de trabajo



Espacios abiertos vs. Despachos individuales

“Coding War Games”

1984-1986, >600 desarrolladores de 92 empresas



	1º cuartil	4º cuartil
Espacio de trabajo	78 sq.ft.	46 sq.ft.
¿Aceptablemente silencioso?	57% sí	29% sí
¿Aceptablemente privado?	62% sí	19% sí
¿Interrupciones innecesarias?	38% sí	76% sí

Diferencias en productividad:

- 10:1 entre individuos (y entre organizaciones)
- 2.6:1 entre el primer y el cuarto cuartil

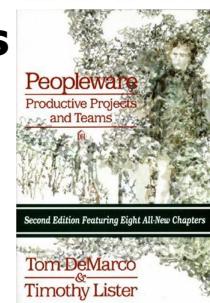


El entorno de trabajo



Espacios abiertos vs. Despachos individuales

- Mentalidad “policial”: Espacios de trabajo diseñados como si fuesen prisiones = Máxima capacidad al mínimo coste.
- Se ignora la productividad (análisis de costes y beneficios sin tener en cuenta los beneficios).



El entorno de trabajo



People cannot work effectively if their workspace is too enclosed or too exposed. A good workspace strikes the balance. . . You feel more comfortable in a workspace if there is a wall behind you. . . There should be no blank wall closer than eight feet in front of you. (As you work, you want to occasionally look up and rest your eyes by focusing them on something farther away than the desk. If there is a blank wall closer than eight feet your eyes will not change focus and they get no relief. In this case you feel too enclosed.) . . . You should not be able to hear noises very different from the kind you make, from your workplace. Your workplace should be sufficiently enclosed to cut out noises which are a different kind from the ones you make. There is some evidence that one can concentrate on a task better if people around him are doing the same thing, not something else. . .

— Christopher Alexander:

“The Timeless Way of Building: A Pattern Language”, 1977



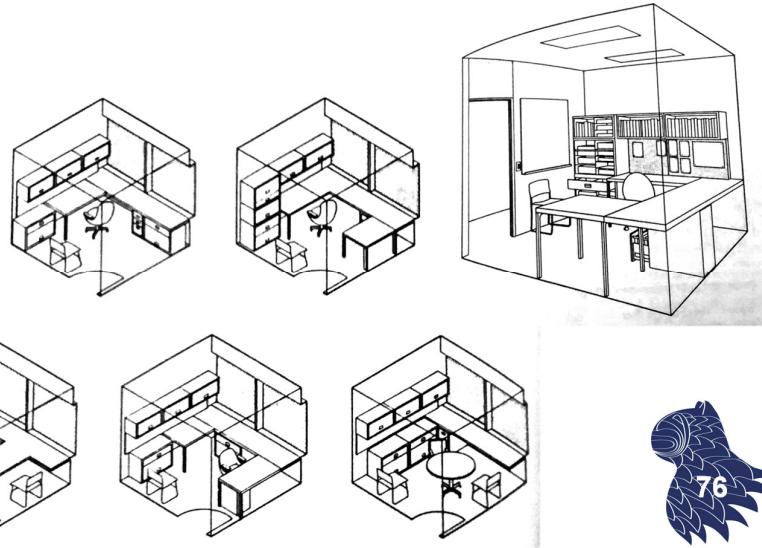
El entorno de trabajo



Espacios abiertos vs. Despachos individuales

Gerald M. McCue: "IBM's Santa Teresa Laboratory—Architectural design for program development"
IBM Systems Journal 17(1):320-341, 1978

Trabajo	Tiempo
Individual	30%
Con otra persona	50%
Con dos o más personas	20%



El entorno de trabajo



Interrupciones

Keep the office quiet.

— Alan M. Davis: "Principles of Software Development"

"The biggest impediment to concentration is your computer's ecosystem of interruption technologies . . . leaving your IM running is like sitting down to work after hanging a giant distract me sign over your desk."

— Cory Doctorow, escritor de ciencia ficción



El entorno de trabajo



Interrupciones

El trabajo intelectual más efectivo se realiza en un estado que los psicólogos denominan “flujo” [flow], completamente inmersos en la tarea

[https://en.wikipedia.org/wiki/Flow_\(psychology\)](https://en.wikipedia.org/wiki/Flow_(psychology))

Task-switching penalty =

- Mechanics of moving to a new task
- + Rework due to inopportune abort
- + Immersion time for think-intensive tasks
- + Frustration (emotional immersion)
- + Loss of team binding effect.

>15 min

-- Tom DeMarco



El entorno de trabajo



Productividad

“There is no such thing as a short-term fix in our business. There is never a way to improve productivity in the short term. . . .

The only real variable you have to play with is the proportion of work hours that are effective. . . .

You have to focus entirely on avoiding wasted time.”

— Tom DeMarco: “The Deadline”, p.79

“There are a million ways to lose a work day, but not even a single way to get one back.”

— Tom DeMarco & Tim Lister: “Peopleware”



El entorno de trabajo



Productividad

$$\text{Factor ambiental} = \frac{\text{Horas sin interrupciones}}{\text{Horas presenciales}}$$

Normalmente, no se está más del 55% del tiempo haciendo trabajo “productivo” en un proyecto.

Otras tareas inevitables requieren nuestra atención: teléfono, correo electrónico, reuniones, emergencias...

Por desgracia, muchos gestores, acostumbrados a trabajar en “modo interrupción”, no son conscientes del impacto que tienen las interrupciones :-(



El entorno de trabajo



Productividad

Prácticas para reducir el número de interrupciones (o, al menos, su impacto en la productividad):

- “No-meeting days”.
- Despachos individuales.
- Señales visibles de tipo “no molestar” [do not disturb].
- Horarios preestablecidos para determinadas tareas.



El entorno de trabajo



Herramientas

Realimentación positiva

"A development team sticks to the programming environment that it knows, only because it allows the team to produce code fast, leading to even more experience with the same programming environment."

-- Gerald M. Weinberg:
Quality Software Management, 1992

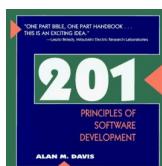


El entorno de trabajo



Herramientas

La técnica antes que las herramientas



Antes de utilizar una herramienta, es necesario tener "disciplina" (comprender y ser capaz de utilizar una técnica apropiada).

Lo que no funciona sin automatizar, tampoco funcionará con automatización.

Chris Kemerer: "How the Learning Curve Affects Tool Adoption", IEEE Software 9(3):23-28, May 1992

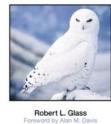


El entorno de trabajo



Herramientas

Facts and Fallacies of Software Engineering



"The most important factor in attacking complexity is not the tools and techniques that programmers use but rather the quality of the programmers themselves."

— **Robert L. Glass:**

"Facts and Fallacies of Software Engineering", 2003.

"Flon's axiom: There does not now, nor will there ever, exist a programming language in which it is the least bit hard to write bad programs."

— **Lawrence Flon:** "On Research in Structured Programming", ACM SIGPLAN Notices, October 1975



El entorno de trabajo



Herramientas

Falsas esperanzas

Facts and Fallacies of Software Engineering



"Most software tool and technique improvements account for about a 5- to 30-percent increase in productivity and quality. But at one time or another, most of these improvements have been claimed by someone to have "order of magnitude" (factor of 10) benefits. Hype is the plague on the house of software."

— **Robert L. Glass:**

"Facts and Fallacies of Software Engineering", 2003.

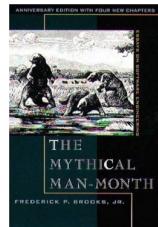


El entorno de trabajo



Herramientas

Falsas esperanzas (a.k.a. “no silver bullets”)



Frederick P. Brooks Jr.:

“No Silver Bullet: Essence and Accidents of Software Engineering”,
IEEE Computer, April 1987



El entorno de trabajo



Cambios

“El cambio es la única constante” — Heráclito

Los sistemas adaptativos complejos revisan y reorganizan sus subsistemas continuamente conforme adquieren experiencia.

[M. Mitchell Waldrop: **Complexity: The Emerging Science at the Edge of Order and Chaos**, 1992]

“If you don’t like change,
you’re going to like irrelevance even less.”

— **General Eric Shinseki**, U.S. Army

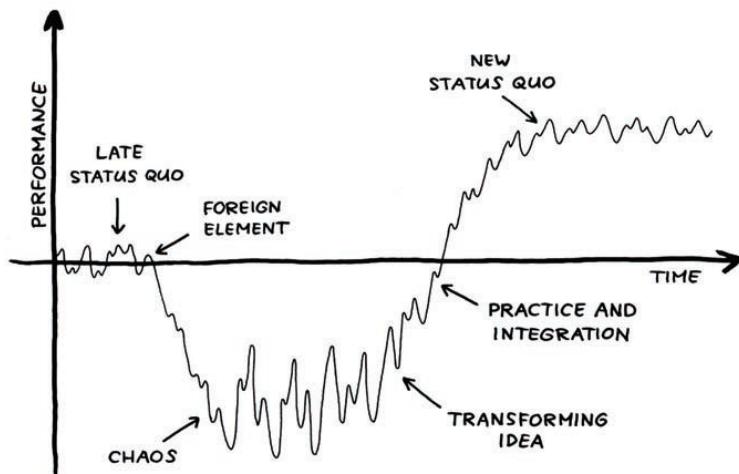


El entorno de trabajo



Cambios

Virginia Satir:
The Satir Model,
1991



Learning a new tool or technique usually **lowers** quality and performance initially, before it goes back up.

-- Robert L. Glass:
"Facts and Fallacies of Software Engineering", 2003.



El entorno de trabajo



Cambios

Henry A. Landsberger:
Hawthorne Revisited, 1958



El efecto Hawthorne

Hawthorne Western Electric Company, 1932

- Aumento de iluminación → Productividad ↑
- Descenso de iluminación → Productividad ↑

Los cambios de productividad pueden producirse como resultado del efecto de los cambios en la motivación de los trabajadores (o del hecho de sentirse observados), no como resultado de los cambios en sí.



Reuniones



There's a temptation in our networked age to think that ideas can be developed by email and iChat. That's crazy. Creativity comes from spontaneous meetings, from random discussions. You run into someone, you ask what they're doing, you say, 'Wow,' and soon you're cooking up all sorts of ideas.

-- Steve Jobs on his strong belief in face-to-face meetings, as quoted in *Steve Jobs* (Simon & Schuster, 2011), by Walter Isaacson, p.431

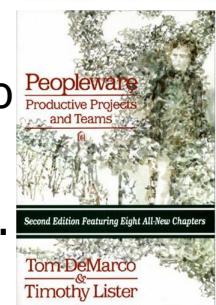


Reuniones



Una reunión debe tener un propósito definido, p.ej. alcanzar una decisión sobre un tema concreto

La reunión debe tener una agenda y ceñirse a ella.



¿Quién debe asistir?

Los que tengan que llegar a un acuerdo antes de que la decisión se considere tomada. Nadie más (y nadie menos).

Si la reunión no se ajusta a estos parámetros, no se trata de una reunión de trabajo...

(el resultado de la reunión no determina su finalización, sino el reloj, en cuyo caso se trata de una “ceremonia”).



Reuniones



- **Stand-up meetings:** breves reuniones al comienzo de cada jornada, usualmente de pie, para facilitar la comunicación cara a cara entre los miembros de un equipo).
- **1:1 ["one-on-ones"]:** Reuniones regulares cara a cara con cada miembro del equipo (p.ej. media hora cada dos semanas), “the single most effective management tool.”
- **Reuniones de 360º,** en las que se realizan evaluaciones de rendimiento en común.



Reuniones



Sobre las políticas de “puertas abiertas”

I can hardly think of a better example of “people management” gone wrong (except perhaps for the phrase “people are our greatest assets”).

-- **Jurgen Appelo: Management 3.0**

- Si los empleados no tienen despachos individuales, las “puertas” indican que no tienen el mismo derecho a su privacidad que los gestores (separación entre clases).
- Incentiva que los empleados pueden “saltarse” a sus superiores inmediatos en la línea de mando (fallos de comunicación y coordinación).



Reuniones



Reuniones en Amazon.com: Sin Powerpoint

Las reuniones comienzan con una lectura en silencio...

"The traditional kind of corporate meeting starts with a presentation. Somebody gets up in front of the room and presents with a **powerpoint** presentation, some type of slide show. In our view you get very little information, you get bullet points. This is **easy for the presenter, but difficult for the audience**. And so instead, all of our meetings are structured around a 6 page narrative memo..." – Jeff Bezos

Brad Porter, Amazon.com VP: **The Beauty of Amazon's 6-Pager**, 2015
<https://www.linkedin.com/pulse/beauty-amazons-6-pager-brad-porter>



Evaluaciones de rendimiento



Deliver more positive than negative feedback.

— Robert Sutton

I praise loudly; I blame softly.

— Catherine the Great of Russia

My job is to not be easy on people.

My job is to make them better.

— Steve Jobs



Evaluaciones de rendimiento



What can companies do to create an atmosphere in which people thrive? Managers can encourage autonomy by letting people make their own decisions about their job. And they can make sure that employees know everything that's going on, because, as they put it, 'Doing your job in an information vacuum is tedious and uninspiring.' Managers should also have zero tolerance for incivility and never allow an employee to poison corporate culture through abuse or disrespect. And, finally, they should give quick and direct feedback.

-- Jeff Sutherland, *Scrum: The Art of Doing Twice the Work in Half the Time*, p.162



Desarrollo profesional



- La rotación en los puestos de trabajo nos hace más flexibles en nuestras actitudes hacia el trabajo de los demás.
- Perspectiva del usuario (p.ej. empresas en las que los ingenieros pasan un tiempo dando soporte técnico).
- Aprendizaje continuo, necesario para construir mejores sistemas (libros, congresos...)
- Asociaciones profesionales: ACM, IEEE...

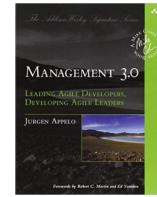


Desarrollo profesional



Formas de ayudar al desarrollo de competencias

Proyectos de desarrollo	Tráfico
Auto-aprendizaje	Cultura
Entrenadores ["coaches"]	Instructores
Certificación	Permiso de conducir
Presión social [peer pressure]	Bocinazos...
Herramientas adaptables	Señales de tráfico
Supervisión	Policía de tráfico
Gestión	Gobierno



Desarrollo profesional



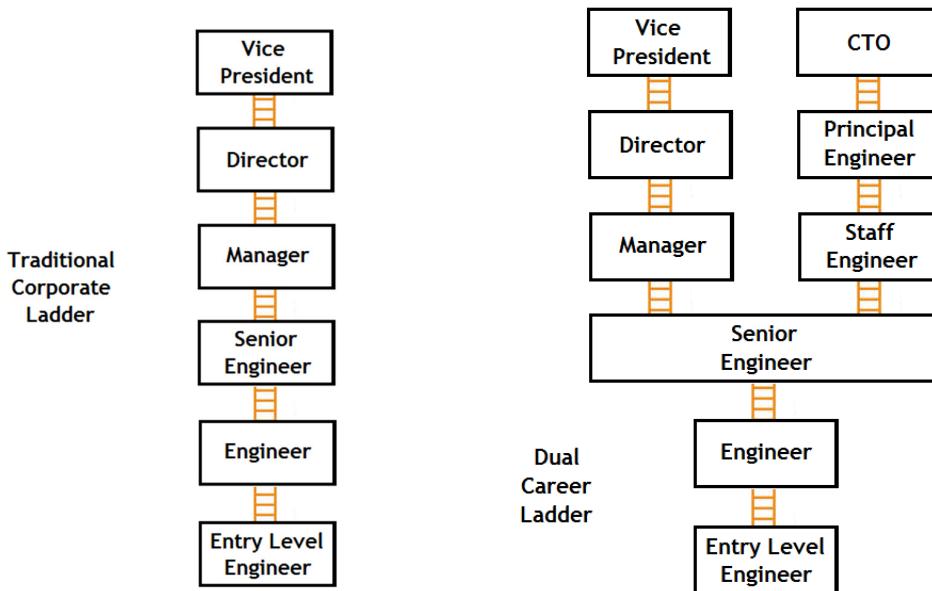
Carrera técnica vs. Carrera de gestión



Desarrollo profesional



Carrera técnica vs. Carrera de gestión



<http://www.computersciencesalaryrange.com/understanding-the-dual-ladder/>



Cultura empresarial



Apple's focus, from the very beginning, has been on the end-user's experience and not on the technology. Steve Jobs famously answered a kind of nasty question at a WWDC more than a decade ago about why he'd killed a particular software product by explaining that the hardest thing about what Apple did was to not try to make a product out of "technology", but to figure out what kind of product they did want to make and then use the technology to do it.

-- David Schlesinger, former Apple networking engineering manager



Apéndice: Código ético



Software Engineering Code of Ethics and Professional Practice

<https://www.acm.org/about/se-code>

Software engineers shall commit themselves to making the analysis, specification, design, development, testing and maintenance of software a beneficial and respected profession.

In accordance with their commitment to the health, safety and welfare of the public, software engineers shall adhere to the following Eight Principles:



Apéndice: Código ético



Software Engineering Code of Ethics and Professional Practice

<https://www.acm.org/about/se-code>

1. **PUBLIC** - Software engineers shall act consistently with the public interest.
2. **CLIENT AND EMPLOYER** - Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
3. **PRODUCT** - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.



Apéndice: Código ético



Software Engineering Code of Ethics and Professional Practice

<https://www.acm.org/about/se-code>

4. **JUDGMENT** - Software engineers shall maintain integrity and independence in their professional judgment.

5. **MANAGEMENT** - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.



Apéndice: Código ético



Software Engineering Code of Ethics and Professional Practice

<https://www.acm.org/about/se-code>

6. **PROFESSION** - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.

7. **COLLEAGUES** - Software engineers shall be fair to and supportive of their colleagues.

8. **SELF** - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

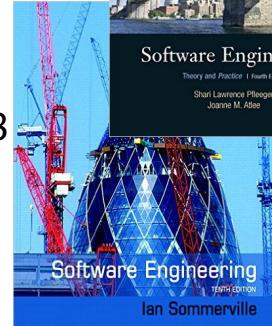
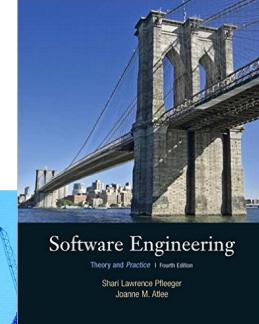
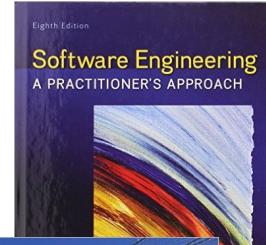


Bibliografía



Libros de texto

- Roger S. Pressman:
Software Engineering: A Practitioner's Approach
McGraw-Hill, 8th edition, 2014. ISBN 0078022126
- Shari Lawrence Pfleeger & Joanne M. Atlee:
Software Engineering: Theory and Practice
Prentice Hall, 4th edition, 2009. ISBN 0136061699
- Ian Sommerville:
Software Engineering
Pearson, 10th edition, 2015. ISBN 0133943038

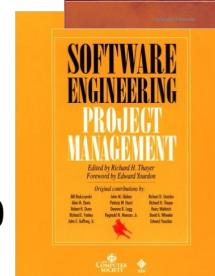
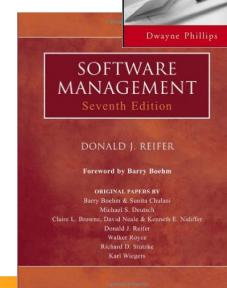


Bibliografía



Lecturas recomendadas

- Dwayne Phillips:
The Software Project Manager's Handbook: Principles That Work at Work
Wiley / IEEE Computer Society, 2nd edition, 2004
ISBN 0471674206
- Donald J. Reifer (editor):
Software Management
Wiley / IEEE Computer Society, 7th edition, 2006
ISBN 0471775622
- Richard H. Thayer (editor):
Software Engineering Project Management
Wiley / IEEE Computer Society, 2nd edition, 2000
ISBN 0818680008

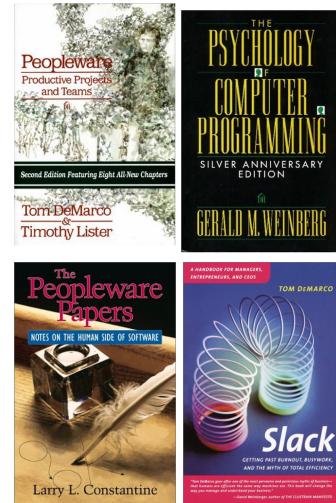


Bibliografía complementaria



Peopleware

- Tom DeMarco & Tim Lister:
Peopleware: Productive Projects and Teams
Dorset House, 2nd edition, 1999.
ISBN 0932633439
Addison-Wesley Professional, 3rd edition, 2013.
ISBN 0321934113.
- Gerald M. Weinberg:
The Psychology of Computer Programming.
Dorset House, Silver Anniversary Edition, 1998.
ISBN 0932633420
- Larry L. Constantine:
The Peopleware Papers: Notes on the Human Side of Software
Prentice-Hall, 2001. ISBN 0130601233
- Tom DeMarco: **Slack: Getting Past Burnout, Busywork, and the Myth of Total Efficiency.** Broadway, 2002. ISBN 0767907698

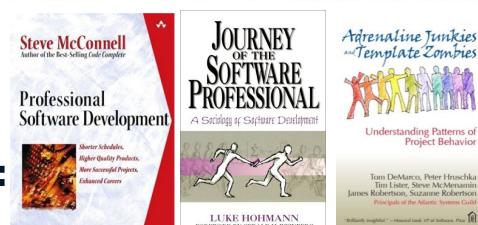


Bibliografía complementaria



Peopleware

- Steve McConnell:
Professional Software Development: Shorter schedules, higher quality products, more successful projects, enhanced careers
Addison-Wesley, 2003. ISBN 0321193679
- Luke Hohmann:
Journey of the Software Professional: The Sociology of Software Development
Prentice-Hall PTR, 1996. ISBN 0132366134
- Peter Hruschka, Tim Lister, Steve McMenamin, James Robertson, Suzanne Robertson & Tom DeMarco:
Adrenaline Junkies and Template Zombies: Understanding Patterns of Project Behaviour
Dorset House, 2009. ISBN 0932633676

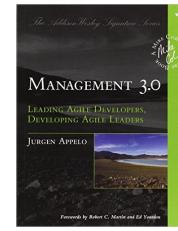
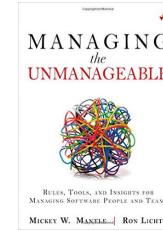


Bibliografía complementaria



Peopleware

- Mickey W. Mantle & Ron Lichty:
Managing the Unmanageable: Rules, Tools, and Insights for Managing Software People and Teams
Addison-Wesley Professional, 2012. ISBN 032182203X
- Jurgen Appelo:
Management 3.0: Leading Agile Developers, Developing Agile Leaders
Addison-Wesley Professional, 2011. ISBN 0321712471
- Josh Tyler:
Building Great Software Engineering Teams: Recruiting, Hiring, and Managing Your Team from Startup to Success
Apress, 2015. ISBN 1484211340

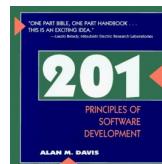
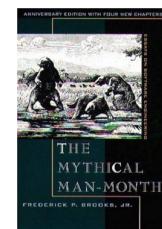


Bibliografía complementaria



Clásicos

- Frederick P. Brooks, Jr.:
The Mythical Man-Month: Essays on Software Engineering
Addison-Wesley, 1995. ISBN 0201835959
- Alan M. Davis:
201 Principles of Software Development
McGraw-Hill, 1995. ISBN 0070158401
- Barry W. Boehm:
Software Engineering Economics
Prentice-Hall PTR, 1991. ISBN 0138221227
- **Manager's Handbook for Software Development**
NASA Software Engineering Laboratory, SEL-84-101, rev.1, 1990.
- **Software Engineering Laboratory (SEL) Relationships, Models, and Management Rules**
NASA Software Engineering Laboratory, SEL-91-001, 1991.

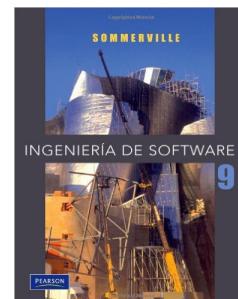


Bibliografía



Bibliografía en castellano

- Roger S. Pressman:
Ingeniería de Software: Un enfoque práctico
McGraw-Hill, 7^a edición, 2010. ISBN 6071503140
- Ian Sommerville:
Ingeniería de Software
Pearson, 9^a edición, 2012. ISBN 6073206038

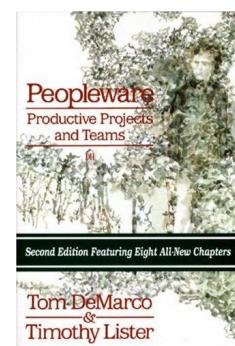


Ejercicios



- **Decálogo de productividad** (prácticas por equipos): Leer el resumen detallado del libro “Peopleware” (<http://javatroopers.com/Peopleware.html>) y proponer un decálogo de medidas concretas que implementaría en su empresa para mejorar la productividad de los equipos de desarrollo de software. Para cada medida propuesta, elaborar una ficha que incluya:

1. Título
2. Descripción (2-3 líneas)
3. Justificación (¿referencias?)
4. Coste



Ejercicios



■ **Procesos de selección de personal:**

Buscar información en Internet sobre el proceso de selección de personal que utilizan, al menos, tres empresas que puedan resultar de su interés. Para cada una de ellas, describir las etapas de las que consta el proceso de selección, el tipo de pruebas al que son sometidos los candidatos y los plazos de tiempo estimados para el proceso completo.

Puede elegir grandes empresas de informática (IBM, HP, Google, Microsoft, Indra...), empresas locales o consultoras (McKinsey, PwC, BCG...).

Sitios como [glassdoor.com](https://www.glassdoor.com) pueden ser de su interés.



Ejercicios



■ **Balance motivacional:**

Factores que motivan (+1) y desmotivan (-1) a una persona concreta dentro en un equipo [>15] (entorno, procesos, prácticas, convenciones establecidas, interacción con los compañeros...).

■ **Valores:** Virtudes clave para un equipo [3-7]

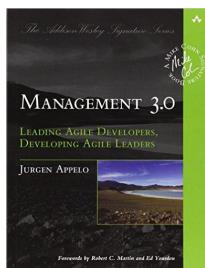
(p.ej. autodisciplina, compromiso, confianza, conocimiento, cooperación, coraje, creatividad, curiosidad, determinación, destreza, entusiasmo, excelencia, exhaustividad, fiabilidad, flexibilidad, franqueza, honestidad, humor, iniciativa, integridad, orden, persistencia, pragmatismo, propósito, racionalidad, resistencia, respeto, responsabilidad, servicio, simplicidad, tacto, tolerancia, transparencia, unidad, visión...)



Ejercicios



Ejemplo de balance motivacional



Ejercicios



Ejemplos de conjuntos de valores

- Programación extrema [XP: eXtreme Programming]: simplicidad, comunicación, realimentación, respeto & coraje
<http://www.extremeprogramming.org/values.html>
- Scrum: atención, coraje, franqueza, coraje, compromiso, respeto
<https://www.scrumalliance.org/why-scrum/core-scrum-values-roles>
- Manifesto for Agile Software Development: individuos e interacciones, software que funcione, colaboración con el cliente, responder a los cambios
<http://www.agilemanifesto.org/>
- Manifesto for Software Craftsmanship: software bien elaborado, añadir valor de forma continua, comunidad de profesionales, colaboración productiva
<http://manifesto.softwarecraftsmanship.org/>

