

Software Engineering (1)

Lec 1

Introduction :

- We can't run the modern world without software.

لا يمكننا تشغيل العالم الحديث بدون برامج . (حياتنا كلها متقدّرش تمشي من غير السوفتوير زي مثلا لو بنتكلم عن حياتنا اليومية احنا لما بنتعامل فالسوبر ماركت بنتعامل مع سوفتوير ببيحسبلي الحاجات اللي اخدتها وبيطلعلي سعر كل حاجة وبيحسبلي الفاتوره بتاعتي ، وكم ان مثلا زي لما بنيجي نتعامل مع كريم او اوبر دول كلهم سوفتوير بيكون ليا عندهم زي سجل او تاريخ يسمحي اني اعمل كومننت اعبر بيه رأيي بيكون مسجل كمان لو ليا فلوس عندهم واقدر استخدمهم فرحله تانيه .. فكل الكلام ده موجود علي سيستم السيستم ده عنده معلومات عن العملاء اللي بيتعاملو معاه دي حاجه من الحاجات الاساسيه اللي بنستخدمها يوميا دي استخداماتنا احنا ، طاب والشركات ؟ او البيزنس عموما كل حاجة فيه محتاجه سوفتوير)

- National infrastructures and utilities are controlled by computer-based systems and most electrical products include a computer and controlling software.

يتم التحكم في البنى التحتية والمرافق الوطنية من خلال أنظمة قائمة على الكمبيوتر ، وتشمل معظم المنتجات الكهربائية جهاز كمبيوتر وبرامج تحكم . (العالم حوالينا كله اصبح عن سوفتوير بنتعامل معاها عشان تسهل علينا تخزين الداتا او استرجاع الداتا او تسهيل التعامل مع اي سيستم تاني)

- Industrial manufacturing and distribution is completely computerized, as is the financial system.

التصنيع والتوزيع الصناعي محوسب بالكامل ، وكذلك النظام المالي .

- Entertainment, including the music industry, computer games, and film and television, is software intensive.

الترفيه ، بما في ذلك صناعة الموسيقى وألعاب الكمبيوتر والأفلام والتلفزيون ، هو برنامج مكثف.

- The software engineering is essential for the functioning of national and international societies.

هندسة البرمجيات ضرورية لعمل المجتمعات الوطنية والدولية . (اصبح كل حاجة حوالينا عبارة عن سوفتوير وبقي في حاجه ملحه لإنشاء وتطوير السوفتوير)

The Purpose of Software Engineering

- The purpose of software engineering is to develop software-based systems that let customers achieve business goals. The customer may be :

الغرض من هندسة البرمجيات هو تطوير أنظمة قائمة على البرامج تتيح للعملاء تحقيق أهداف العمل. قد يكون العميل :

(لما نتكلم عن الهدف الاساسي للسوفتوير فالحاجه الاساسيه اللي احطها فدماعي هي كلمه "ازاي ارضي احتياجات العميل ؟ " سواء كان العميل ده ..)

- a hospital manager who needs patient-record software to be used by secretaries in doctors' offices; or

مدير مستشفى يحتاج إلى برنامج لتسجيل المرضى ليتم استخدامه من قبل السكرتارية في مكاتب الأطباء ؛ أو

- a manufacturing manager who needs software to coordinate multiple parallel production activities that feed into a final assembly stage.

مدير تصنيع يحتاج إلى برنامج لتنسيق العديد من أنشطة الإنتاج المتوازية التي تدخل مرحلة التجميع النهائية .

- Software engineer must understand the customer's business needs and design software to help meet them.

يجب أن يفهم مهندس البرمجيات احتياجات عمل العميل وتصميم البرامج للمساعدة في تلبيتها .
(كل عميل سيكون لديه طلبات معينة فلازم اهتم بمتطلبات كل واحد واحتياجاته وكمال لازم هو يحس بعد لما اخصله السيستم اللي عاوزه اني فعلا حققتله كل متطلباته اللي هو عاوزها)

- **This task requires the following:** تتطلب هذه المهمة ما يلي

- The ability to quickly learn new and diverse disciplines and business processes.

القدرة على التعلم السريع للتخصصات والعمليات التجارية الجديدة والمتنوعة .

(يعني لازم مهندس البرمجيات يكون عنده القدره انه يفهم كل مجال ومتطلباته واساسياته ع حسب السيستم المطلوب منه ، مثلا احنا بنعمل سيستم لناس كثير جدا بمجالات مختلفه اكيد مش هنبقي فاهمين كل حاجه فيها ، مثلا احنا مش متخصصين فالطب ولا البيزنس فبنلاقهم بيتكلموا فحاجات احنا مش فاهمينها فلازم نفهم المبادئ المختلفه لكل مجال عشان احنا اللي هنترجمها).

- The ability to communicate with domain experts, extract an abstract model of the problem from a stream of information provided in discipline-specific jargon, and formulate a solution that makes sense in the context of customer's business.

القدرة على التواصل مع خبراء المجال ، واستخراج نموذج تجريدي للمشكلة من تدفق المعلومات المقدمة بلغة تخصصية محددة ، وصياغة حل يكون منطقيًا في سياق عمل العميل .
(لازم يكون عندنا قدره وصبر علي العملاء وعلي اني اسمع واحلل عشان اقدر اطلع المتطلبات الحقيقيه والقدرة علي التواصل مع خبراء المجال عشان اقدر اوصل للهدف اللي العميل عاوز يوصله وعشان اقدر اعرف المشكله واشوف الحلول المناسبه ليها)

- The ability to design a software system that will realize the proposed solution and gracefully evolve with the evolving business needs for many years in the future.

القدرة على تصميم نظام برمجي يحقق الحل المقترح ويتطور بأمان مع احتياجات العمل المتطورة لسنوات عديدة في المستقبل .

Developing ATM Software تطوير برمجيات أجهزة الصراف الآلي

- an example of designing a software system to operate an automatic banking machine, known as Automatic Teller Machine (ATM).

مثال على تصميم نظام برمجي لتشغيل آلة مصرفية آلية ، تُعرف باسم ماكينة الصراف الآلي ATM .

- We know that an ATM machine allows us to deposit or withdraw money, and we can imagine how to split these activities into simpler activities to be performed by imaginary little “agents” working inside the machine.

نحن نعلم أن ماكينة الصراف الآلي تسمح لنا بإيداع الأموال أو سحبها ، ويمكننا تخيل كيفية تقسيم هذه الأنشطة إلى أنشطة أبسط يقوم بها "وكلاء" صغرون وهميون يعملون داخل الجهاز .

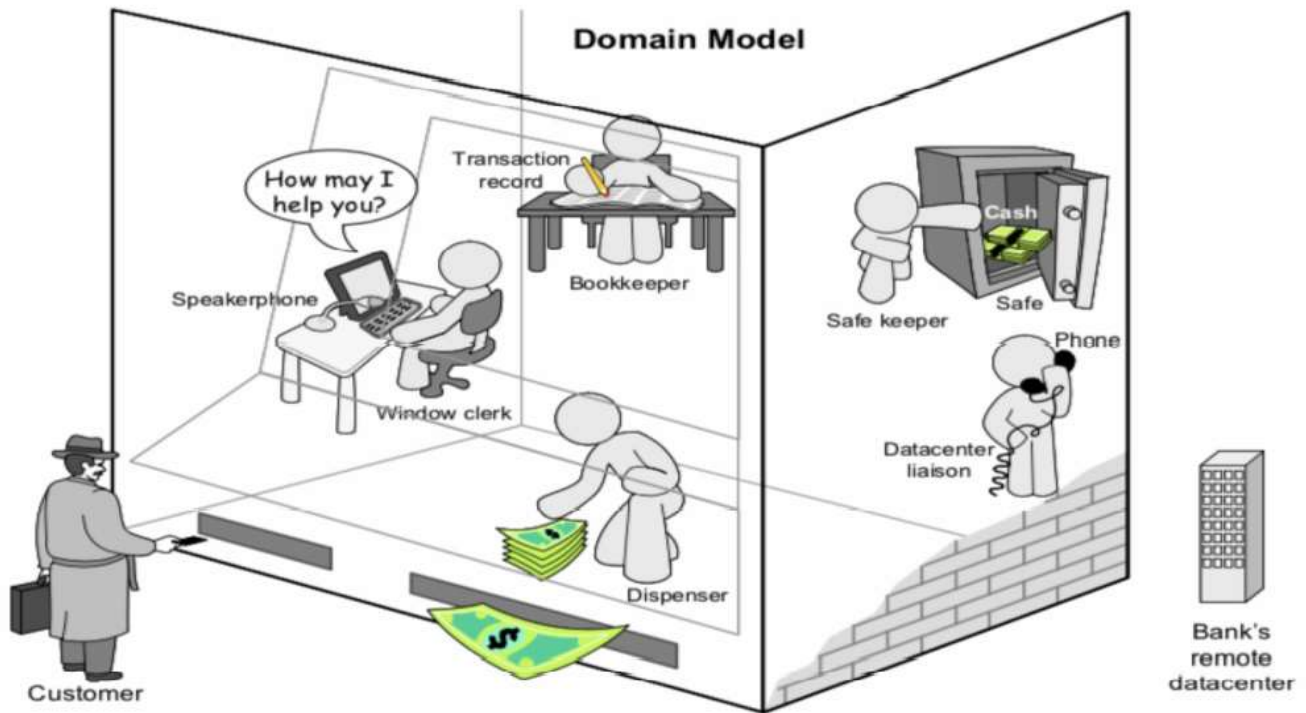


Figure 1-4: Imagined static structure of ATM shows internal components and their roles.

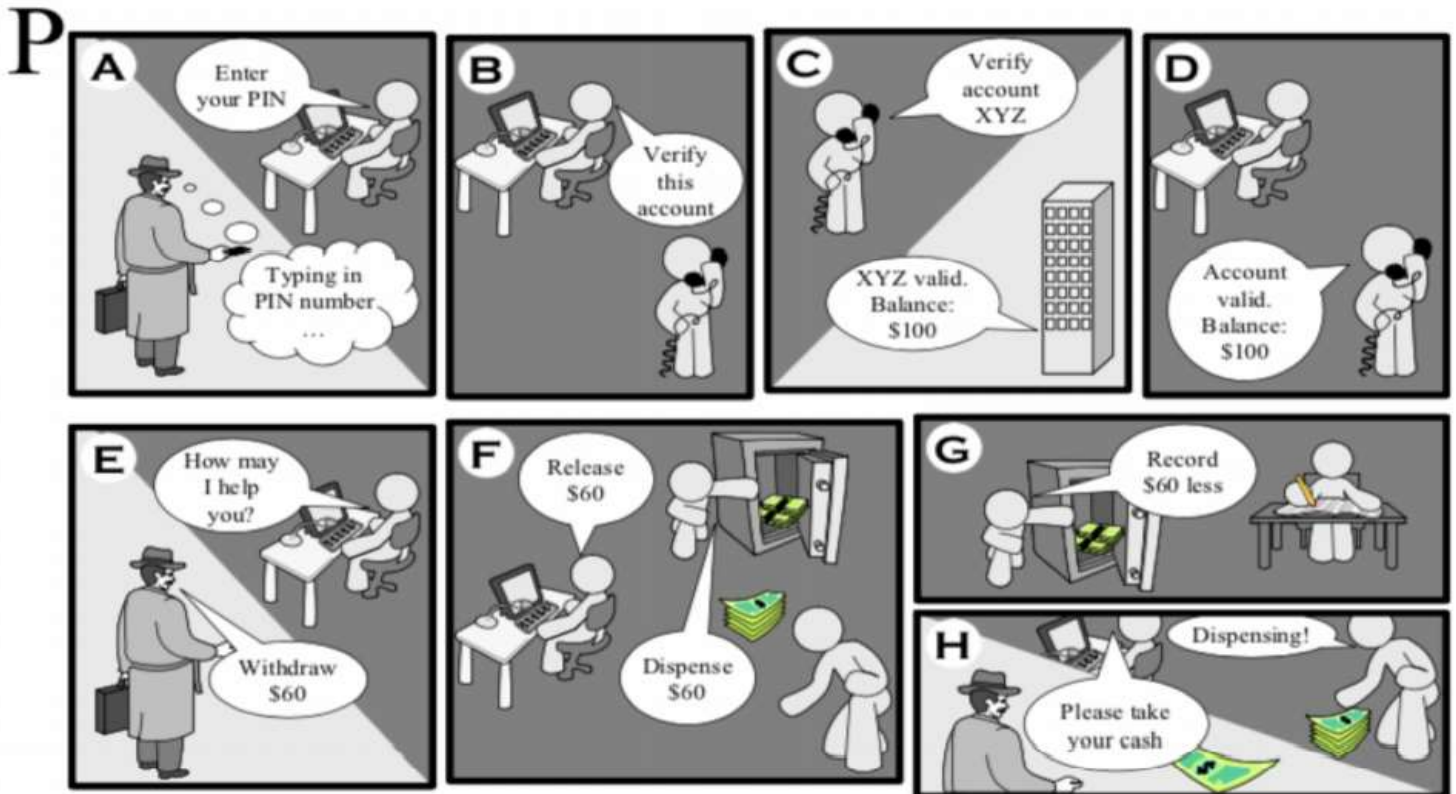


Figure 1-5: Dynamic interactions of the imagined components during task accomplishment.

What is the Software Engineering ? ما هي هندسة البرمجيات ؟

- The term software engineering is the product of two words, software, and engineering.

مصطلح هندسة البرمجيات هو نتاج كلمتين ، البرمجيات والهندسة .

- The Software is more than just a program code. A program is an executable code, which serves some computational purpose. Software is considered to be a collection of executable programming code, associated libraries and documentations. Software, when made for a specific requirement is called software product.

البرنامج أكثر من مجرد رمز برنامج. البرنامج عبارة عن رمز قابل للتنفيذ ، يخدم بعض الأغراض الحسابية. يعتبر البرنامج مجموعة من كود البرمجة القابل للتنفيذ والمكتبات والوثائق المرتبطة به. يُطلق على البرنامج ، عند إعداده لمتطلبات معينة ، منتج البرنامج .

- Engineering is all about developing products, using well-defined, scientific principles and methods, design, build, maintain, and improve frameworks, processes, etc.

تدور الهندسة حول تطوير المنتجات ، واستخدام المبادئ والأساليب العلمية المحددة جيدًا ، والتصميم ، والبناء ، والصيانة ، وتحسين الأطر والعمليات ، وما إلى ذلك .

- So, we can define software engineering as an engineering branch associated with the development of software product using welldefined scientific principles, methods and procedures.

لذلك ، يمكننا تعريف هندسة البرمجيات على أنها فرع هندسي مرتبط بتطوير منتج البرمجيات باستخدام مبادئ وأساليب وإجراءات علمية محددة جيدًا .

- The outcome of software engineering is an efficient and reliable software product.

نتيجة هندسة البرمجيات هي منتج برمجي فعال وموثوق .



THE CONFUSION BETWEEN SOFTWARE ENGINEERING& PROGRAMMING

• Software Engineering

هندسة البرمجيات

- Software engineering is the creative activity of understanding the business problem, coming up with an idea for solution, and designing the “blueprints” of the solution.

هندسة البرمجيات هي النشاط الإبداعي لفهم مشكلة العمل ، والتوصل إلى فكرة للحل ، وتصميم "مخططات" الحل .

- Software engineer's focus is on understanding the interaction between the system-to-be and its users and the environment, and designing the software-to-be based on this understanding.

يركز مهندسو البرمجيات على فهم التفاعل بين النظام المستقبلي ومستخدميه والبيئة ، وتصميم البرامج على أساس هذا الفهم .

• Programming

- Programming is the craft of implementing the given blueprints

البرمجة هي حرفة تنفيذ المخططات المحددة .

- programmer's focus is on the program code and ensuring that the code faithfully implements the given design.

ينصب تركيز المبرمج على كود البرنامج والتأكد من أن الكود ينفذ التصميم المحدد بأمانة .

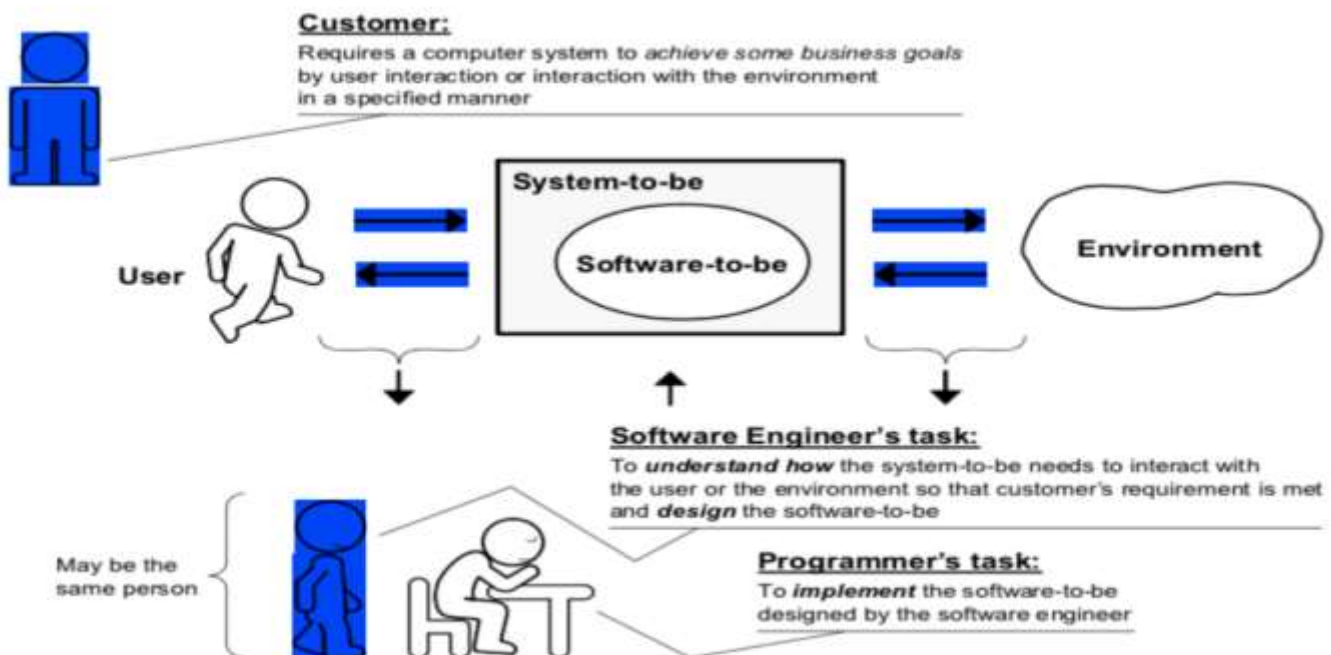


Figure 1-1: The role for software engineering.

أهمية هندسة البرمجيات **THE IMPORTANCE of Software Engineering**

- Without using software engineering principles it would be difficult to develop large programs.

بدون استخدام مبادئ هندسة البرمجيات ، سيكون من الصعب تطوير برامج كبيرة

- A problem with developing such large commercial programs is that the complexity and difficulty levels of the programs increase exponentially with their sizes.

مشكلة تطوير مثل هذه البرامج التجارية الكبيرة هي أن مستويات التعقيد والصعوبة للبرامج تزداد أضعافاً مضاعفة مع أحجامها

- Software engineering helps to reduce this programming complexity. تساعد هندسة البرمجيات على تقليل تعقيد البرمجة هذا .
- Software engineering principles use two important techniques to reduce problem complexity: abstraction and decomposition.

تستخدم مبادئ هندسة البرمجيات تقنيتين مهمتين لتقليل تعقيد المشكلة: التجريد والتحلل .

- **Abstraction:** The main purpose of abstraction is to consider only those aspects of the problem that are relevant for certain purpose and suppress other aspects that are not relevant for the given purpose.

التجريد: الغرض الرئيسي من التجريد هو النظر فقط في جوانب المشكلة ذات الصلة لغرض معين وقمع الجوانب الأخرى غير ذات الصلة بالغرض المحدد (يعني من الآخر فكرتها ان انا اخص المشكله في فقره بسيطه تبين اي هي المشكله بشكل مفهوم وتبين اي الهدف من ان انا احلها واي كل النواحي المرتبطه بيها عشان خاطر احلها واشيل التفاصيل الزياده اللي ملهاش لازمه .. اهم حاجه اني اوصل واعرف اي هي المشكله عشان اعرف احلها).

- **Decomposition:** a complex problem is divided into several smaller problems and then the smaller problems are solved one by one.

التحلل: يتم تقسيم المشكلة المعقدة إلى عدة مشاكل أصغر ثم يتم حل المشكلات الأصغر واحدة تلو الأخرى . (هقسم المشكله بتاعتي لشويه مشاكل صغيره عشان اقدر احل كل واحده منهم الاول لوحدها).

أهداف هندسة البرمجيات The Objectives of Software Engineering

- The need of software engineering arises because of higher rate of change in user requirements and environment on which the software is working.

تنشأ الحاجة إلى هندسة البرمجيات بسبب ارتفاع معدل التغيير في متطلبات المستخدم والبيئة التي يعمل عليها البرنامج .

1. Reduces Complexity: يقلل من التعقيد:

It is easier to build a wall than to a house or building, likewise, as the size of software become large engineering has to step to give it a scientific process. Large software systems are always complicated and challenging to progress. Software engineering divides big problems into various small issues; and then start solving each small issue one by one. All these small problems are solved independently to each other and then integrated together to produce the software product.

من الأسهل بناء جدار بدلاً من بناء منزل أو مبنى ، بالمثل ، حيث أن حجم البرنامج الذي أصبح هندسيًا كبيرًا يجب أن يتقدم لإعطائه عملية علمية. دائمًا ما تكون أنظمة البرامج الكبيرة معقدة وصعبة التقدم. تقسم هندسة البرمجيات المشاكل الكبيرة إلى قضايا صغيرة متنوعة ؛ ثم ابدأ في حل كل مشكلة صغيرة واحدة تلو الأخرى. يتم حل كل هذه المشكلات الصغيرة بشكل مستقل عن بعضها البعض ثم يتم دمجها معًا لإنتاج منتج البرنامج .
(كل حازه لو اتقسمت حتي لو حازه صغيره لو اتقسمت لمجموعات او مجموعات من المشاكل والتحديات وبدأنا نبص عليها ونطلع حلول لكل واحد علي حدي وبعدين نجمع كل الكلام ده مع بعض فالحاله دي هنقدر نوصل لسيستم يكون موثوق فيه وسيستم يكون فعال .

2. To minimize software cost: لتقليل تكلفة البرنامج

Software requires a lot of hard work and software engineers are highly paid professionals. A lots of man force is requires to develop software with millions of codes. But in software engineering, programmers plan everything and reduce all those things that are not required. In turn, cost for software productions becomes less as compared to any software that does not use software engineering approach.

يتطلب البرنامج الكثير من العمل الجاد ومهندسي البرمجيات هم محترفون يتقاضون رواتب عالية. يتطلب الأمر الكثير من القوة البشرية لتطوير برامج بملايين الرموز. لكن في هندسة البرمجيات ، يخطط المبرمجون لكل شيء ويقللون من كل تلك الأشياء غير المطلوبة ، في المقابل ، تصبح تكلفة إنتاج البرامج أقل مقارنة بأي برنامج لا يستخدم نهج هندسة البرمجيات .

3. To decrease time: لتقليل الوقت

- One of the main criteria to measuring project success and yet the most challenging is delivering software projects on time. Schedule issues are the main reason for conflicts on projects, especially during the second half of projects where the actual implementation takes place to produce the final working product. Software engineering involves the processes required to ensure timely completion of a project.

أحد المعايير الرئيسية لقياس نجاح المشروع ومع ذلك فإن التحدي الأكبر هو تقديم مشاريع البرمجيات في الوقت المحدد. قضايا الجدول هي السبب الرئيسي للتعارض على المشاريع ، خاصة خلال النصف الثاني من المشاريع حيث يتم التنفيذ الفعلي لإنتاج منتج العمل النهائي. تتضمن هندسة البرمجيات العمليات المطلوبة لضمان الانتهاء من المشروع في الوقت المناسب . (الوقت هو عبارته عن Cost.. منظمه جدا للوقت والتاسكات فكره اننا نستخدم مبادئ هندسه البرمجيات والمراحل بتاعتها اللي بتخلينا نتأكد بعد كل فتره معينه من اننا خلصنا التاسك كذا مثلا).

4. Reliable software :

برنامج موثوق

• Software should be reliable, means if you have delivered the software then it should work for at least it's given time span or subscription, and if any bugs come in the software then company is responsible for solving all these bugs. Because in software engineering, testing and maintenance is provided so there is no worry of its reliability.

يجب أن يكون البرنامج موثوقًا ، بمعنى أنه إذا قمت بتسليم البرنامج ، فيجب أن يعمل على الأقل لفترة زمنية محددة أو اشتراك ، وإذا ظهرت أي أخطاء في البرنامج ، فإن الشركة مسؤولة عن حل جميع هذه الأخطاء. لأنه في هندسة البرمجيات ، يتم توفير الاختبار والصيانة لذلك لا داعي للقلق بشأن موثوقيتها. (عشان نعمل اسم لشركتنا مينفعش نطلع بروجيكت من عندنا اي كلام)

• **Testability**—The software can be easily tested. This generally means that the software is written in a modular manner.

قابلية الاختبار - يمكن اختبار البرنامج بسهولة. هذا يعني عمومًا أن البرنامج مكتوب بطريقة معيارية .

• **Maintainability**—The software can be easily understood and changed over time if problems occur. This term is often used to describe the lifetime of long-lived systems such as the air traffic control system that must operate for decades.

قابلية الصيانة - يمكن فهم البرنامج بسهولة وتغييره بمرور الوقت في حالة حدوث مشكلات. غالبًا ما يستخدم هذا المصطلح لوصف عمر الأنظمة طويلة العمر مثل نظام التحكم في الحركة الجوية الذي يجب أن يعمل لعقود .

5. Reusability—Some or all of the software can be used again in other projects. This means that the software is modular, that each individual software module has a well-defined interface, and that each individual module has a clearly defined outcome from its execution. This often means that there is a substantial level of abstraction and generality in the modules that will be reused most often.

قابلية إعادة الاستخدام — يمكن استخدام بعض البرامج أو كلها مرة أخرى في مشاريع أخرى. هذا يعني أن البرنامج معياري ، وأن كل وحدة برمجية فردية لها واجهة محددة جيدًا ، وأن كل وحدة فردية لها نتيجة محددة بوضوح من تنفيذها. هذا يعني غالبًا أن هناك مستوى كبير من التجريد والتعميم في الوحدات التي سيتم إعادة استخدامها في أغلب الأحيان .

6. Effectiveness:

فعالية

• Better process of software development provides better and quality software product. Effectiveness means getting the desired results/ doing the right thing! From example: achieving missions and goals; generating satisfied customers; producing work of high quality. his includes proper use of : communication, technology, organizational and individual knowledge, and resources

توفر عملية تطوير البرامج الأفضل منتجًا برمجيًا أفضل وعالي الجودة. الفعالية تعني الحصول على النتائج المرجوة / القيام بالشيء الصحيح! على سبيل المثال: تحقيق المهام والأهداف ؛ توليد رضا العملاء ؛ إنتاج أعمال ذات جودة عالية. يتضمن الاستخدام السليم لما يلي: الاتصالات والتكنولوجيا والمعرفة التنظيمية والفردية والموارد .
(لازم تكون النتائج اللي طالعه هي النتائج الصحيحه هي بتحقق فعلا افكار ومتطلبات وهدف العميل وترضيه).

TYPICAL SOFTWARE ENGINEERING TASKS المهام النموذجية لهندسة البرامج

There are several tasks that are part of every software engineering project:

هناك العديد من المهام التي تشكل جزءًا من كل مشروع هندسة برمجيات

- Analysis of the problem تحليل المشكلة
- Determination of requirements تحديد المتطلبات (طلبات العميل)
- Design of the software تصميم البرنامج
- Coding of the software solution كتابه اكواد الحل البرمجي (يحقق التارجت)
- Testing and integration of the code اختبار وتكامل الكود
- Documentation توثيق (تثبيت)
(مهم جدا حتي وانا بكتب برنامج لازم اكتب كومننت جامب كل فانكشن او كلاس احنا بنعمله واقول اي اللي بيتم واي الهدف من الفانكشن والتاسك مهم جدا ليا عشان لو نسيت الحاجه دي بتعمل اي وعشان لو سببت الشركه وجه حد مكاني ميتوهش ويقدر يفهم الدنيا ويكمل بسهولة).
- Maintenance صيانة
- Quality assurance ضمان الجودة
- Training تمرين (تجربه)

Software Engineering Ethics

أخلاقيات هندسة البرمجيات

- Like other engineering disciplines, software engineering is carried out within a social and legal framework that limits the freedom of people working in that area.

مثل التخصصات الهندسية الأخرى ، يتم تنفيذ هندسة البرمجيات ضمن إطار اجتماعي وقانوني يحد من حرية الأشخاص العاملين في هذا المجال .

- As a software engineer, you must accept that your job involves wider responsibilities than simply the application of technical skills.

بصفتك مهندس برمجيات ، يجب أن تقبل أن وظيفتك تنطوي على مسؤوليات أكبر من مجرد تطبيق المهارات التقنية .

- You must also behave in an ethical and morally responsible way if you are to be respected as a professional engineer.

يجب أيضاً أن تتصرف بطريقة أخلاقية ومسؤولة أخلاقياً إذا كنت تريد أن تحظى بالاحترام كمهندس محترف .

- It goes without saying that you should uphold normal standards of honesty and integrity.

وغني عن القول أنه يجب عليك الالتزام بالمعايير العادية للأمانة والنزاهة .

(يعني مثلاً لو بشتغل فبروجيكت خاص بالضرايب مبدأياً عشان اكون أمين لازم التزم بقوانين وحدود الدولة يعني مينفعش اعمل سيستم يخلي العميل يتهرب من الضرايب مش بس عشان خايف من المسئولية القانونية لا وكمان عشان دي اخلاقيات المهنة الاساسيه)

- You should not use your skills and abilities to behave in a dishonest way or in a way that will bring disrepute to the software engineering profession. However, there are areas where standards of acceptable behavior are not bound by laws but by the more tenuous notion of professional responsibility.

يجب ألا تستخدم مهاراتك وقدراتك للتصرف بطريقة غير شريفة أو بطريقة تضر بسمعة مهنة هندسة البرمجيات. ومع ذلك ، هناك مجالات لا تلتزم فيها معايير السلوك المقبول بالقوانين ولكن بالمفهوم الأكثر هشاشة للمسؤولية المهنية .

(يعني أراعي الامانه لو عندي اضافات مثلاً من الامانه اني اقولها له حتي لو كانت رؤيته محدوده ومن الامانه انك تعرض عليه شكل افضل للسيستم مش اريح دماغي واقول هو عاوز الحته دي فعملها له وخلاص لا اعرضها عليه وهو ورغبته سواء وافق او رفض) .

Some of those ethics are:

بعض هذه الأخلاق هي

- **Confidentiality** You should normally respect the confidentiality of your employers or clients irrespective of whether or not a formal confidentiality agreement has been signed.

السرية يجب أن تحترم عادة سرية أصحاب العمل أو العملاء بغض النظر عما إذا كان قد تم توقيع اتفاقية سرية رسمية أم لا . (السريه حتي لو انا شايف ان المعلومات دي مش سريه يعني مثلا لو العميل قالي فكره كريتيك عشان اضيفها للبروجيكت بتاعه ونفذتهاله مش من حقي اعرضها علي عميل فبروجيكت ثاني قبل م ينشرها عشان اكسب مينفعش والموضوع هيبقي اكبر لما يكون ع مستوي الدوله احنا عندنا كمان درجات فالسريه)

- **Competence** You should not misrepresent your level of competence.

You should not knowingly accept work that is outside your competence.

الكفاءة يجب ألا تحرف مستوى كفاءتك. يجب ألا تقبل عمدا عملا خارج اختصاصك .
(من باب الامانه لما اكون شايف اني مش هقدر اعمل البروجيكت واحققه بنسبه ١٠٠% معمولوش حتي لو هو فرصه)

- **Intellectual property rights** You should be aware of local laws governing the use of intellectual property such as patents and copyright. You should be careful to ensure that the intellectual property of employers and clients is protected.

حقوق الملكية الفكرية يجب أن تكون على دراية بالقوانين المحلية التي تحكم استخدام الملكية الفكرية مثل براءات الاختراع وحقوق النشر. يجب أن تكون حريصًا على ضمان حماية الملكية الفكرية لأصحاب العمل والعملاء

- **Computer misuse** You should not use your technical skills to misuse other people's computers. Computer misuse ranges from relatively trivial (game playing on an employer's machine, say) to extremely serious (dissemination of viruses or other malware).

إساءة استخدام الكمبيوتر يجب ألا تستخدم مهاراتك التقنية لإساءة استخدام أجهزة الكمبيوتر الخاصة بالآخرين. تتراوح إساءة استخدام الكمبيوتر من التافهة نسبيًا (اللعب على جهاز صاحب العمل ، على سبيل المثال) إلى الخطورة للغاية (نشر الفيروسات أو البرامج الضارة الأخرى)