

## AGILE SOFTWARE

اللَّهُمَّ صَلِّ عَلَى مُحَمَّدٍ وَعَلَى آلِ مُحَمَّدٍ، كَمَا صَلَّيْتَ عَلَى إِبْرَاهِيمَ، وَبَارِكْ عَلَى مُحَمَّدٍ وَعَلَى آلِ مُحَمَّدٍ،  
كَمَا بَارَكْتَ عَلَى آلِ إِبْرَاهِيمَ، فِي الْعَالَمِينَ، إِنَّكَ حَمِيدٌ مَجِيدٌ

- Rapid development and delivery are now often the most important requirement for software systems.
- Agile aim was to radically reduce the delivery time for working software systems.

## AGILE DEVELOPMENT

- Program specification, design and implementation are inter-leaved. (incremental model). مبني كده انه
  - The system is developed as a series of versions or increments with stakeholders involved in version specification and evaluation.
  - Deliver working software quickly.
  - Minimal documentation focuses on working code.
- أول حاجه نعرفها ان Agile شغال بمبدأ incremental بمعنى انا مش هسلمك المشروع مره واحده لا ان بدل ما أسلم السيستم كله علي بعض لأن انا هسلم الأول جزء من السيستم فيه العناصر الرئيسيه بس وبعد كده هضيف العناصر الفرعية و اسلمهولك. هنا برضه بيهم بالكد أكثر من Documentation يعني بدل ميعمل Documentation في شهر لا هعملها في أسبوع و 3 أسابيع دول أكتب فيهم كود للسيستم

- A plan-driven approach to software engineering is based around separate development stages with the outputs to be produced at each of these stages planned in advance.

هنا بيحصل التكرار في كل مرحله لحد ما أخلصها زي Waterfall Model

⇒ Iteration occurs within activities.

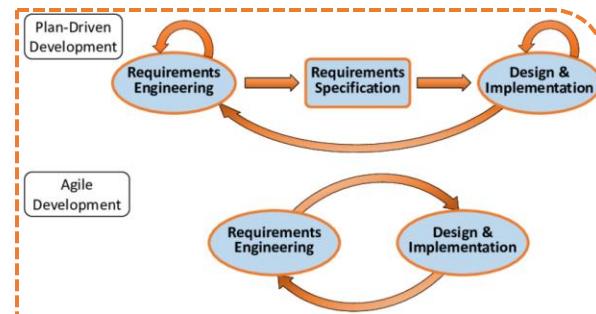
### Agile development:

⇒ Specification, design, implementation and testing are inter-

leaved. ينتقل من مرحله للتنبيه عادي من غير ملخص اللي انا فيها و ممكن

رجع أعدل في المرحله اللي فاتت .

⇒ Iteration occurs across activities.



### AGILE METHODS:

#### These methods:

- Focus on the **code** rather than the **design**. زي مقولنا أهم حاجه عندي هي الكود

- Are based on an **iterative** approach to software development. سيكون في دايما عمل تطوير للسيستم

- Are intended to **deliver working software quickly**. بيهتم بتسلیم السيسم بأسرع وقت

#### The aim of agile methods is to reduce overheads in the software process.

#### The principles of agile methods:

- Customer involvement

- Incremental delivery

- People not process

- Embrace change

- Maintain simplicity

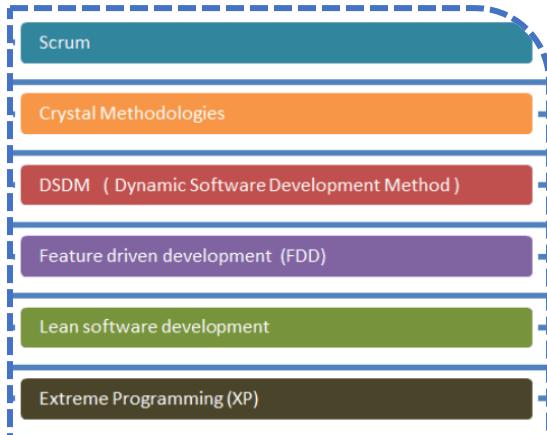


اللهم اجعل لأهل فلسطين النصرة  
والعزة والغلبة والقوة والهيبة

## ❖ Agile method applicability:

- A small or medium-sized product for sale. **يستخدم في البرامج الصغيرة او المتوسطه**
- Custom system development within an organization, where there is a clear commitment from the customer to become involved in the development process. **عشان تطور السيستم لازم العميل يشارك في التطوير ده عن طريقه انه يديك فيدبك**

## ❖ AGILE PROCESS MODELS:



## AGILE DEVELOPMENT TECHNIQUES

### ✳ EXTREME PROGRAMMING:

- Increments are delivered to customers every 2 weeks. **بتسلم تطوير للسيسم كل أسبوعين**
- All tests must be run for every build and the build is only accepted if tests run successfully.

**انت لما بتعمل تحديث علي السيستم لازم تعمل test للسيستم كله عشان تتأكد ان الجزء اللي صفتة مبوظش السيستم.**

### ✳ User stories for requirements:

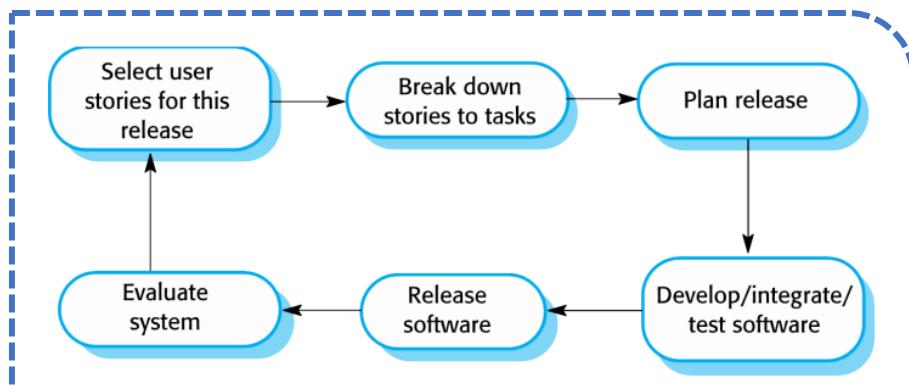
#### ➤ In Extreme Programming (XP):

- ⇒ a customer or user is part of the XP team and is responsible for bringing system requirements to the team for implementation. **انت بتعتبر اليوزر فرض من التيم اللي شغال علي المشروع لانه هو اللي بيقولك علي المتطلبات**
- ⇒ User requirements are expressed as user stories or scenarios. **بحاول اوصاف متطلبات اليوزر في شكل قصص**
- ⇒ These are written on cards and the development team break them down into implementation tasks. **انت خدت المتطلبات علي شكل استوري بعد كده انت بتقسم المتطلبات دي الي تاسكت**

- ⇒ The customer chooses the stories for inclusion in the next release based on their priorities.

**العميل و هو بيقولك علي المتطلبات ضروري انه يحدلك المتطلبات الاوليه او ذات أهميه أكبر.**

### ❖ The extreme programming release cycle:



**اللهم اجعل لأهل فلسطين النصرة  
والعزة والغلبة والقوة والهيبة**

## ❖ USER STORY:

- **User story:** A brief, simple requirement statement from a user's perspective.

### USER STORY TEMPLATE

Story Title	User ID
User Story 1	Importance Estimate
As a .....(stakeholder) I want to .....(task), So That .....(desired result)	Type
Acceptance Criteria	Payment Report/view Search Manage data Workflow
Measurable results, what defines "done"? And I know I am done when	

اول حاجه عندنا **User Story** و دي الصيغه اللي انا بناء عليه باخد المتطلبات من العميل.

اما بقى **Acceptance Criteria** و ده بيكون عباره عن المعايير اللي مفروض تبقي موجوده في السيستم علشان يقبل بها العميل

## ❖ INVEST IN GOOD USER STORIES:

- **Independent:** User Stories should be as independent as possible. في شكل قصه مش لازم علشان مش قولنا بنأخذ الطلبات عميلين عندهم نفس فكرة السيسم يكون ليهم نفس القصه لازم يكون كل عميل له قصه منفصله بيها.
- **Negotiable:** a User Story is not a contract. It is not a detailed specification. It is a reminder of features for the team to discuss and collaborate to clarify the details near the time of development. القصه مبيكونش فيها تفاصيل او هي
- **Valuable:** User Stories should be valuable to the user (or owner) of the solution. They should be written in user language. They should be features, not tasks.
- **Estimable:** The user stories should have sufficient detail for the dev team to understand and estimate them.
- **Small:** User Stories should be small. Not too small and not too big. لازم تكون قصيره لأننا بنكتب كعنواين مش بالتفصيل.
- **Testable:** User Stories need to be worded in a way that is testable. This is supported by the "Confirmation".

## ❖ REFACTORING:

انت لما تعمل سистем و تسلمه للعميل مهو أكيد السيستم ده ه يحتاج تطوير بعد كده فكرة انك تعمل **improvements** لكود ده بنقول عليها **refactoring** و أكيد علشان تعدل على كود و تطور فيه لازم تكون فاهمه كويس جدا فعلشان كده لازم الكود بتاعك يكون **well-structured** بحيث لو حد من التيم اللي معاك مفروض يعدل فيه يكون فهم الكود سهل و بناء عليه يكون التعديل فيه سهل.

### ➤ EXAMPLES OF REFACTORING:

- ✓ Re-organization of a class hierarchy to remove duplicate code.
- ✓ Tidying up and renaming attributes and methods to make them easier to understand.

## ❖ Test-first development:

انت أول ما تعمل أول **Version** من السوق وير لازم طبعا تعملها **test** وأي تعديل بتعمله في الكود اللي أصلا بيكون الغرض منه التطوير لازم تعمل **Test** بحيث تتأكد ان الجزء اللي صفتة او عدلتة مبوظش الدنيا. هل انت كديفيلوبر اللي بتعمل **Test**? الاجابه لا لأن مستحيل انك لوحدك تكتشف كل الأخطاء اللي ممكن تظهر في السيستم علشان كده بنقول ان اليوزر همان بيكون له دور رئيسي و فعال في عملية **Testing**

- **Testing is central to XP and the program is tested after every change has been made.**

اللهم اجعل لأهل فلسطين النصرة والعزة والغلبة والقوة والهيبة

#### ❖ PAIR PROGRAMMING:

عارفين طبعاً ان مفيش واحد بس اللي بيشغل على الكود لا ده بيكون تيم كامل و كل شخص بيشتغل على جزء . هنا بقى في **pair programming** انا هخلி كل 2 بيشتغلوا على نفس الجزء فهيدلث sharing of knowledge اللي عند كل واحد فيهem و بناء عليه هيطلعوا كود أفضل . طب افرض واحد من التيم مشي عادي مش هنحضر نوقف السيسystem لان 2 كانوا شغالين على الجزء ده فممكن واحد يكمل و كده أكون امنت نفسي من مشكلة اني أوقف السيسystem علشان في واحد من التيم مشي . وده برضه بيشعج التيم انه دائمًا يعمل **refactoring** لسيستم .

• في عندي مصطلح ثاني اسمه **Collective ownership** بمعنى انا مش هقى على كل واحد في التيم تاسك او جزء من السيسystem لأى شخص في التيم ممكن يشتغل على أي جزء في الكود و يعدل فيه .

• في مصطلح ثاني اسمه **on-site customer** و ده العميل بيكون فرد في التيم بمعنى هو مطلوب منه دائمًا انه يبلغ التيم بأى متطلبات بحيث ان التيم يعملها **implementation** و يكون دائمًا متاح في أي وقت التيم يسأله على حاجة في requirements .

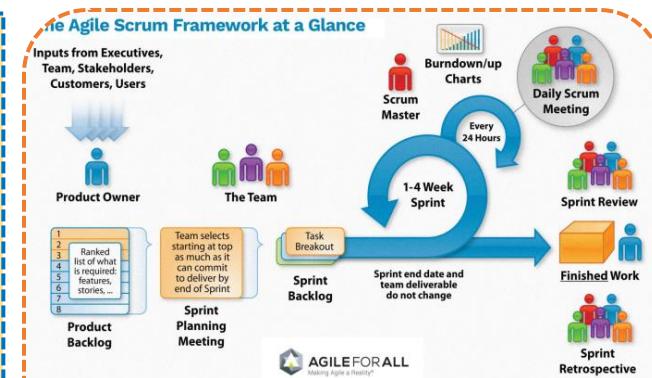
## AGILE PROJECT MANAGEMENT

بيكون عندنا شخص اسمه **project manager** و ده بيكون مسئول انه يعمل **plan-driven** للسيستم بمعنى يكون حاطط ميعاد تسليم اول Version للعميل بيكون برضه مسئول عن تحديد الميزانية اللي هتصرف على السيستم لحد ما يخلاص و هكذا .

#### ❖ SCRUM:

➤ **Scrum** is an agile method that *focuses on managing iterative development*.

ال Scrum **يتنقسم الى 3 مراحل** و دي التيم **يبقعد فيها مع العميل** و يأخذ منه المتطلبات اللي هو عاوزه و **يحددوا المتطلبات الأساسية** و بناء على الخطوه دي **يبقرروا ازاى هينفذوا المشروع** و ده **يكون عباره عن مجموعة من التطويرات اللي انا هعملها على sprint cycles** و ده **السيستم** و **sprint** الواحدة **بتاخذ من أسبوع الى 4 أسابيع** . و في المرحله دي انت بتكون خلاص بتخلص في المشروع و بتكون على وشك الانتهاء من كل المتطلبات اللي في **.documentation**



ال **Product Owner** **يبكون شخص او مجموعه من الأشخاص** **يحددوا المتطلبات اللي هتكون في السوق** وير و **prioritize these for** و **يراجعوا product backlog** باستمرار لضمان انه **يتحقق** الطلبات اللي هما عاوزينها .

➤ The Product Owner can be a customer but might also be a **product manager**.

#### ✚ Product Backlog:

⇒ **This is a list of 'to do' items which the Scrum team must develop.** **يبكون** فيها مجموعة المتطلبات اللي مفروض **تنفذ**

⇒ **Definitions for the software requirements.**

#### ✚ Team: **A self-organizing group** of software developers, which **should be no more than 7 people**.

✚ **A daily meeting of the Scrum team** that reviews progress and prioritizes work to be done that day. Ideally, this should be a short face-to-face meeting that includes the whole team.

ال Scrum master يكون مسئول عن تحديد ميعاد الميتاج اللي بيتعمل كل يوم لل팀 اللي شغال علي السوق وير . بيتبع السوق وير و يشوف مدي التقدم اللي اتحقق و ايه اللي لسه مخلصش من backlog بيحدد القرارات اللي التيم مفروض يعملها و يتواصل مع العملاء .

#### ❖ SPRINT CYCLE:

- Sprint are a development iteration. Sprints are usually 1-4 weeks long.
- The starting point for planning is the product backlog, which is the list of work to be done on the project.
- selection phase involves all of the project team who work with the customer to select the features and functionality from the product backlog to be developed during the sprint.

## REQUIREMENTS ENGINEERING

- ❖ Requirements: are the descriptions of the system services and constraints that are generated during the requirements engineering process. يكون وصف لي احنا عاوزينه يكون موجود في السيستم
- ❖ Requirements Engineering Process: The process of establishing the services that the customer requires from a system and the constraints under which it operates and is developed. ازاي نطبق المتطلبات

Functional requirements	Non-functional requirements
Specific services or functions the system must provide	Define system properties and constraints.
must build into the product to enable users to accomplish their tasks.	Define system properties and constraints. ( performance , reliability ,security ,usability)
الى اليوzer يكون طالبها في السيستم بتاعه Function بتحتوي على	هنا احنا بنحدده من الناحية التقنية شويع يعني لازم يكون سريع و نوع OS اللي هيستغل عليه و ايه هي اللغة اللي هنكتبه بيهها
Example: search the appointments lists , ID for users ..	Example: Product requirement, Organizational requirement, External requirement

- ❖ Non-functional requirements may be more critical than functional requirements. If these are not met, the system may be useless.

#### ❖ Metrics for specifying nonfunctional requirements:

- Speed:  
⇒ Processed transactions/second. ⇒ User/event response time.
- Ease of use:  
⇒ Training time. ⇒ Number of help frames.
- Reliability:  
⇒ Mean time to failure. ⇒ Rate of failure occurrence.
- Robustness:  
⇒ Time to restart after failure. ⇒ Probability of data corruption on failure.  
⇒ Percentage of events causing failure.

- ❖ Goal: A general intention of the user such as ease of use.

- ❖ Verifiable non-functional requirement: A statement using some measure that can be objectively tested.

اللهم اجعل لأهل فلسطين النصرة  
والعزة والغلبة والقوة والهيبة

User requirements	System requirements
Abstract Points	More Detailed Description
Services and constraints of the system <u>in abstract</u>	Services and constraints of the system <u>in detail</u>
Readable by everybody	Useful for the design and development Precise and cover all cases

❖ Requirements engineering processes:

- The process of establishing the services that the customer requires from a system and the constraints under which it operates and is developed.

أهنا أخذنا الطلبات من العميل بنادها بعد كده نعملها analysis ونشوف ازاي تنفذ كل طلب من دول في السيستم بحيث انه يحقق متطلبات اليوزر بالإضافة الى Security و القيود اللي انا مفروض بضميفها للسوفت وير .

- There are a number of generic activities common to all processes:
  - ⇒ Requirements elicitation and analysis.
  - ⇒ Requirements specification.
  - ⇒ Requirements validation.
  - ⇒ Requirements management.
- In practice, RE is an iterative activity in which these processes are interleaved.

اللهم اجعل لأهل فلسطين النصرة والعزة والغلبة والقوة والهيبة

اللَّهُمَّ صَلِّ عَلَى مُحَمَّدٍ وَعَلَى آلِ مُحَمَّدٍ، كَمَا صَلَّيْتَ عَلَى إِبْرَاهِيمَ، وَبَارِكْ عَلَى مُحَمَّدٍ وَعَلَى آلِ مُحَمَّدٍ،  
كَمَا بَارَكْتَ عَلَى آلِ إِبْرَاهِيمَ، فِي الْعَالَمِينَ، إِنَّكَ حَمِيدٌ مَحِيدٌ