# تصور مقترح لإدارة الجامعات المصرية
# فى ضوء مداخل تطوير إدارة عمليات البرمجيات


إعداد

أ. د /أمانى السيد غبور

أستاذ أصول التربية

كلية التربية النوعية - جامعة المنصورة



مجلة بحوث التربية النوعية ـ جامعة المنصورة

عدد (٦١) ـ يناير ٢٠٢١

# تصور مقترح لإدارة الجامعات المصرية في ضوء مداخل تطوير إدارة عمليات البرمجيات

إعداد

أ . د /أماني السيد غبور *

## الملخص

لقد ظهرت كثير من المداخل والاتجاهات الحديثة في الإدارة وتم تطبيق كثير منها فى المصانع والشركات منذ سنوات طويلة ولكنه لم يتم الحديث عنها أو تطبيقها فى إدارة المؤسسات التعليمية إلا في السنوات القليلة الماضية. وانطلاقاً من تحديات القرن الحالي على المستويين العالمي والإقليمي التي أدت إلى حدوث تحولات جذرية متسارعة في كل أوجه الحياة طالت عدة جوانب محورية فى مؤسسات التعليم العالي، إضافة إلى ما أحدثته ثورة المعلومات والاتصالات من مفاهيم جديدة. فضلاً عما تواجهه الجامعات المصرية من التحديات والتي تفرض عليها ضرورة استحداث أساليب تفكير جديدة في الإدارة بشكل عام والإدارة الجامعية بشكل خاص؛ ومحاولة وضع تصورات مستقبلية للإدارة الجامعية يكون أساسها التميز والإبداع واستشراف المستقبل، وإعادة النظر في فلسفتها ورسالتها وسياستها واستراتيجياتها وخططها وبرامجها التعليمية وكوادرها البشرية وهياكلها التنظيمية، وتقوية الروابط بينها وبين المجتمع، إلى جانب تحقيق التفاعل الإيجابي بينها وبين ما يجري حولها من تغيرات وتحولات جذرية.

وعلى ضوء أهمية تطوير البرمجيات (إدارة عمليات البرمجيات)، وانطلاقاً من اهتمام الباحثة بضرورة تطوير الأداء بمؤسسات التعليم العالي؛ فقد سعى البحث نحو الوقوف على الإطار المفاهيمي لهندسة البرمجيات وأهميتها ومعوقاتها، وتطويرها (إدارة عمليات البرمجيات) وتطورها في الفكر الإداري المعاصر، والتعرف على أهم التحديات التي تواجه هندسة البرمجيات، مع عرض لبعض النماذج الشائعة فى تطوير البرمجيات (إدارة عمليات البرمجيات)، ووصولاً لتقديم تصور مقترح لإدارة الجامعات المصرية فى ضوء مداخل تطوير إدارة عمليات البرمجيات، معتمداً على المنهج الوصفي للتعرف على الأسس النظرية لهندسة البرمجيات، مع توضيح أهم مداخل تطوير إدارة عمليات البرمجيات وصولاً للتصور المقترح.

## مقدمة

لقد أدت تحديات القرن الحالي على المستويين العالمي والإقليمي إلى حدوث تحولات جذرية متسارعة في كل أوجه الحياة، إضافة إلى ما أحدثته ثورة المعلومات والاتصالات من مفاهيم جديدة.

* أستاذ أصول التربية كلية التربية النوعية - جامعة المنصورة

وقد طالت تلك التحولات والمفاهيم عدة جوانب محورية فى مؤسسات التعليم العالي، الأمر الذي تطلب ضرورة استحداث أساليب تفكير جديدة في الإدارة بشكل عام والإدارة الجامعية بشكل خاص؛ حيث بات واضحاً أن العالم كله يتوجه في إدارته المستقبلية نحو تنظيمات ذات سمات تنظيمية إدارية جديدة.

ويواجه التعليم الجامعي مجموعة من التحديات أبرزها: (١)

- العولمة والمنافسة العالمية التى أدت إلى تغيير مسار حركة التعليم الجامعي نتيجة للشروط الجديدة التي فرضتها على كل الدول.
- النهوض بالتعليم لتحقيق حاجات ومتطلبات المجتمع.
- الثورة المعلوماتية بما قدمته من منجزات علمية وتكنولوجية كان لها أثر كبير في تزايد الفجوة بين الدول. وأصبحت إدارة المعرفة جزءًا لا يتجزء من فلسفة التعليم وهو أحد أكثر الاتجاهات الأسرع تطورًا والتى تزيد من تنافسية التعليم العالي فى عالم اليوم (٢).

كما غدا الحاسوب مكون رئيس في جميع نواحي الحياة خاصة المجال المعلوماتي ومجال قواعد البيانات؛ حيث أصبح من الصعب حصر الشركات المصنعة للبرمجيات التي تسهل الأعمال المكتبية والإدارية في الشركات الخاصة والعامة والبنوك والمنظمات الحكومية والعلمية والجامعات، من خلال تسهيل العلميات التي تجري لأي بيانات متداولة بكثرة والمتكررة في تلك المنظمات بسرعة كبيرة جداً ودقة عالية، لذا أصبحت البرمجيات العنصر الجوهري في تطور النظم والمنتجات المعتمدة على الحاسوب.

وفى إطار هذا الوضع الجديد، تواجه الجامعات المصرية تحديات كثيرة تفرض عليها ضرورة وضع تصورات مستقبلية للإدارة الجامعية يكون أساسها التميز والإبداع واستشراف المستقبل وإعادة النظر في فلسفة تلك المؤسسات، ورسالاتها، وسياساتها، واستراتيجياتها، وخططها، وبرامجها الأكاديمية، وكوادرها البشرية، وهياكلها التنظيمية، والأهداف التي تسعي إلى تحقيقها، وكيفية تحقيق التفاعل الإيجابي بينها وبين ما يحيط بها من تغيرات وتحولات جذرية، وتقوية الروابط بينها وبين المجتمع.

## مشكلة البحث

تعد الجامعات من أهم المؤسسات التي تقوم على إنتاج المعرفة والاستثمار فيها، وهي من أكثر المؤسسات ملاءمة لتبني إدارة المعرفة. وتؤكد العديد من الدراسات أن تبني إدارة المعرفة يحقق عدداً من الفوائد منها: تطور ونمو المنظمات، وتحسين عملية اتخاذ القرارات، وتحقيق الميزة التنافسية، وتحسين الإبداع وسرعة الاستجابة، وزيادة الإنتاجية، وخفض التكاليف، وزيادة الكفاءة والفعالية وتحسين الأداء (٣ ، ٤ ، ٥).

وتواجه الجامعات المصرية مجموعة كبيرة من التحديات، والتي تفرض عليها أن تغير من طبيعتها وأسلوب عملها التقليدي، سواء من ناحية الإدارة أو التعليم أو الأساليب والتقنيات أو

الهياكل والبني الجامعية أو الأهداف وطرق التقويم والتعامل مع المجتمع وتزويده بالمهارات العلمية المدربة.

وقد ظهرت كثير من المداخل والاتجاهات الحديثة في الإدارة وتم تطبيق كثير منها فى المصانع والشركات منذ سنوات طويلة ولكنه لم يتم الحديث عنها أو تطبيقها فى إدارة المؤسسات التعليمية إلا في السنوات القليلة الماضية، ومن بين هذه المداخل والاتجاهات:

- الإدارة التحويلية Transformational Management
- الإدارة الإستراتيجية StrategicManagement
- الإدارة العنكبوتية SpiderManagement
- إدارة رأس المال الفكريIntellectualCapitalManagement
- إدارة الجودة الشاملةTotalQualityManagement
- إدارة المعرفةKnowledgeManagement
- إدارة الأزماتCrisisManagement
- إدارة الاجتماعات MeetingManagement
- إدارة التغييرChangeManagement
- إدارة التميزExcellenceManagement
- إدارة الكفاءات(Competencymanagement (Talentmanagement
- إدارة الوقتTimeManagement
- الإدارة الالكترونيةElectronicManagement
- الإدارة الإستشرافية FutureManagement
- الإدارة المرئيةVisualManagement
- الإدارة بالذكاء العاطفيEmotionalIntelligenceManagement
- الإدارة الوظيفيةFunctionalManagement

وانطلاقاً من الواقع الحالي الذى يشير إلى أنه قد بات لزامًاعلى كل المنظمات أن تسعى إلى امتلاك نظاماً إدارياً قوياً وفعالاً يساعدها على البقاء والاستمرار، من خلال التكيف مع بيئة الأعمال الجديدة، وإفرازات العولمة، وحدة التنافسية فى عالم سريع التغير؛ فلابد أن تضطلع الجامعات بدورها في إحداث تغييرات كيفيه في سياساتها والأساليب المتبعة في إدارتها، وأن تستفيد من مواردها البشرية في تصميم وتطبيق مستجدات الفكر الإداري لتطوير أدائها، ولما كان مدخل تطوير إدارة عمليات البر مجيات يعد من الأساليب الحديثة التي دخلت مجال الإدارة بصفة عامة وهندسة البر مجيات بصفة خاصة؛ فإنه يمكن تحديد تساؤلات البحث الحالي فيما يلي:

**كيف يمكن إدارة الجامعات المصرية فى ضوء مداخل تطوير إدارة عمليات البر مجيات؟**

ويمكن أن يتفرع عن هذا التساؤل عدة تساؤلات فرعية على النحو التالي:

١. ما مفهوم هندسة البر مجيات؟ وما أهميتها في الفكر الإداري المعاصر؟

٢. ما أهم منهجيات تطوير إدارة عمليات البر مجيات؟

٣. ما معوقات تطوير إدارة عمليات البر مجيات ؟

٤. ما التصور المقترح لإدارة الجامعات المصرية في ضوء مداخل تطوير إدارة عمليات البر مجيات؟

## أهمية البحث

إن التنافس بين مؤسسات التعليم العالي – لاسيما بعد زيادة عدد المؤسسات الخاصة التي تقدم الخدمة التعليمية – يفرض عليها ضرورة تبنى آليات تؤدى إلى تطوير أدائها، الأمر الذي يؤدى إلى تجويد خدماتها وتلبية حاجات المجتمع. من هنا تتحدد الأهمية النظرية والتطبيقية للبحث الحالي في الجوانب التالية:

١. يعد نقطة انطلاق نحو دراسات مستقبلية في مصر لمدخل جديد فى إدارة الجامعات.

٢. تأتي مسايرة للاهتمام المتزايد بتطوير عمليات إدارة البر مجيات بوصفها نهجاً إدارياً حديثاً وأثره في تطوير الأداء بالجامعات.

٣. صلتها بموضوع تكنولوجيا المعلومات، وهو محور تطوير العملية الإدارية في ظروف الانفتاح والمنافسة والعولمة.

٤. الحاجة الماسة في الجامعات للتحديث والتطوير لمواجهة التحديات المستقبلية.

٥. يمكن أن تستفيد من نتائجه الجامعات الحكومية والخاصة في مصر.

٦. محدودية البحوث والدراسات التي اهتمت بهذا المدخل في إدارة الجامعات في العالم العربي عامة ومصر خاصة في حدود علم الباحثة.

٧. محاولة لتقديم مدخلاً جديداً لإدارة الجامعات المصرية في ضوء مداخل تطوير إدارة عمليات البر مجيات قد تفيد المسئولين والقائمين عليها بتزويدهم بمجموعة من الاستراتيجيات والخطوات الاجرائية للتطوير.

## أهداف البحث

سعى هذا البحث إلى تحقيق الأهداف التالية:

١. توضيح مفهوم هندسة البر مجيات وأهميتها في الفكر الإداري المعاصر.

٢. التعرف على أهم منهجيات تطوير إدارة عمليات البر مجيات.

٣. التعرف على معوقات تطوير البر مجيات.

٤. وضع "تصور مقترح" لإدارة الجامعات المصرية في ضوء مداخل تطوير إدارة عمليات البر مجيات.

## مصطلحات البحث

**هندسة البرمجيات** software Engineering

تعرف إجرائياً بأنها "مجموعة الاستراتيجيات والمنهجيات والأدوات اللازمة لبناء أنظمة برمجية عالية الجودة وفق الوقت المحدد والجودة المطلوبة والميزانية المخصصة لها".

**تطوير البرمجيات** (**إدارة عمليات البرمجيات**)

تعرف إجرائياً بأنها "أسلوب إدارى تستخدمة المؤسسات وفرق مشاريع البر مجيات لتطبيق إطار عمل بهدف التخطيط والتنظيم والسيطرة على عملية تطوير نظام معلوماتى ما".

## منهج البحث

استخدمت الباحثة في تناولها لموضوع البحث المنهج **الوصفي** نظرًا لملاءمته لطبيعة الدراسة؛ من خلال ما توافر للباحثة من كتابات ودراسات سابقة متعلقة بمجال البحث. بهدف التعرف على الأسس النظرية لهندسة البر مجيات، مع توضيح أهم مداخل تطوير إدارة عمليات البر مجيات. وصولاً للتصور المقترح لإدارة الجامعات المصرية في ضوء مداخل تطوير إدارة عمليات البر مجيات.

## خطوات البحث

للإجابة عن تساؤلات هذا البحث وتحقيق أهدافه، سوف يتم التركيز على المحاور التالية:

- أولاً: هندسة البر مجيات (المفهوم – الأهمية – المعوقات).
- ثانياً: مداخل تطوير إدارة عمليات البر مجيات
- ثالثاً: التصور المقترح.

**أولاً: هندسة البرمجيات** (**المفهوم – الأهمية – المعوقات**)

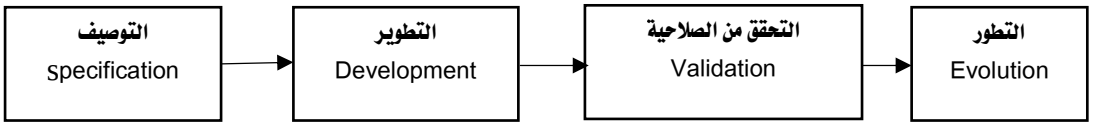
هندسة البر مجيات software Engineering هي الاختصاص الهندسي الذي يهدف إلى توفير إطار عمل لإنتاج برمجيات ذات جودة عالية؛ فتعرف بأنها مجموعة التقنيات والمنهجيات والأدوات التي تساعد على بناء أنظمة برمجية عالية الجودة ضمن ميزانية معطاة وخلال فترة زمنية محددة رغم التطورات المستمرة وأثناء التطوير. (٦)

كما تعرف أيضا بأنها "مجموعة من الأساليب والاستراتيجيات والخطط لإنشاء برامج ذات كفاءة عالية جدًا وفق الوقت المحدد والجودة المطلوبة والميزانية المتفق عليها. (٧)

وتختلف هندسة البر مجيات software Engineering عن علم الحاسوب computer scienceفى أنها تركز على الجوانب العملية المتعلقة ببناء البر مجيات المفيدة وتسليمها. أماعلم الحاسوب يهتم بدراسة النظريات والمنهجيات التي تشكل الأساس في بناء النظم الحاسوبية والبر مجية. كما تختلف أيضًا عن هندسة النظم system engineering؛ فهندسة النظم تهتم

بجميع الجوانب المتعلقة بتطوير النظم المعتمدة على الحاسوب بما فيها من تصميم وبناء للتجهيزات والبرمجيات ومن عناية بإجراءات التطوير. أما هندسة البرمجيات فهي تهتم بتطوير البنية البرمجية الأساسية والتحكم والتطبيقات والمعطيات التي تدخل في تكوين هذا النظام. وتعتبر هندسة النظم أقدم من هندسة البرمجيات، ومع تزايد نسبة البرمجيات في الأنظمة أصبح من الضرورى الاهتمام بهندسة البرمجيات بشكل خاص. خاصة الإجرائية البرمجية software process وهي مجموعة الأنشطة التي تهدف إلى تطوير البرمجيات وتطورها ومهما اختلف المختصون على تفاصيل هذه الأنشطة، فإنهم متفقون على وجود الأنشطة الأربعة التالية: (٨)

١. التوصيف specification: ويهدف إلى تحديد ما يجب على النظام فعله وشروط التطوير.
٢. التطوير Development: ويهدف إلى إنتاج النظام البرمجي.
٣. التحقق من الصلاحية Validation: وتهدف إلى التأكد من أن البرمجية الناتجة هي فعلا الشئ الذي كان المستخدم ينشده.
٤. التطور Evolution: ويعني بالتعديلات على البرمجية استجابة للتغيرات في المتطلبات.




### طرق هندسة البرمجيات software engineering methods (٩)

نطلق طريقة على مجموع نماذج النظام وطرق التدوين والقواعد والنصائح التصميمية وتوجيهات الإجرائية التي تعتمدها منهجية ما في تطوير البرمجيات؛ فلم تظهر أطر عمل منهجيات تطوير البرمجيات حتي الستينيات من القرن العشرين، وبحسب إليوت (Ellott, 2004) فإن دورة حياة تطوير البرمجيات، يمكن أن يعتبر أقدم منجية قد تم تشكيلها لبناء الأنظمة المعلوماتية، والفكرة الرئيسة في دورة حياة تطوير البرمجيات كانت في تطوير الأنظمة المعلوماتية بطريقة منظمة ومتبعة للمنهجية العلمية، مما يتطلب من كل مرحلة من مراحل دورة حياة التطوير أن تضع فكرة تسليم النظام النهائي نصب عينيها. وتوجد العديد من الطرق في هندسة البرمجيات ففي السبعينيات من القرن الماضى ظهرت الطرق الموجهة بالوظيفة مثل طريقة التحليل البنيوى Structured Analysis التي وضعها DeMarco وطريقة JSD التي وضعها Jackson وفى الثمانينيات والتسعينيات من القرن نفسه ظهرت الطرق الموجهة بالأغراض مثل طريقه Booch وRumbaugh. ولكل طريقة من هذه الطرق فلسفتها وتوصيفها ونماذجها المعتمدة ومخططاتها البيانية وقواعدها وتوجيهاتها وترتيبها الخاص بها لأنشطة التطوير.

### دورة حياة تطوير البرمجيات (إدارة عمليات البرمجيات)

تمر دورة حياة تطوير البرمجيات بخمس مراحل رئيسية هى (١٠):

١. المتطلبات Requirement
٢. التصميم Design
٣. التطوير Development
٤. الفحص Testing
٥. الصيانة Maintence

**معوقات تطوير البرمجيات (١١)**

في السنوات الأولي لتطوير البرمجيات، كانت معظم متطلبات المستخدمين مستقرة إلى حد ما، وعملية تطوير البرمجيات تتبع الخطط دون تغييرات كبيرة ومع ذلك، بما أن تطوير البرمجيات ينطوي على مشاريع صناعية أكثر حيوية وديناميكية؛ فقد ظهرت صعوبات جديدة، وفقا لنمو الشركات وتشمل هذه الصعوبات:

- المتطلبات المتطورة: إن متطلبات العملاء تتغير بسبب احتياجات الأعمال المتطورة معظم العملاء ليس لديهم رؤية واضحة حول مواصفات متطلباتهم في المراحل المبكرة يدرك بعض العملاء متطلباتهم الحقيقية فقط عندما يستخدمون تطبيقًا لا يلبي احتياجاتهم حقا، مصدر أخر للتغيير يأتي من الخبرات المكتسبة أثناء التطوير.
- مشاركة العملاء: يؤدي عدم مشاركة العملاء إلى زيادة فرص فشل المشروع؛ فبعض الشركات عادة لا تخصص أي جهد لمشاركة العملاء.
- المواعيد النهائية والميزانيات: لا يقبل العملاء الفشل في كثير من الأحيان، من ناحية أخري، تقدم الشركات عادة ميزانيات منخفضة، ومواعيد نهائية صارمة، بينما في الوقت نفسه تتطلب متطلبات عالية، وكل ذلك بسبب المنافسة في الأسواق.
- سوء الفهم: أحد أسباب سوء الفهم للمتطلبات هو سوء الفهم بين المطورين والعملاء، وهذا يؤدي إلى سوء فهم احتياجات العميل.

وتكمن أهم التحديات التي تواجه هندسه البرمجيات في القرن الواحد والعشرين في النقاط الثلاث التالية: (١٢)

١- عدم التجانس: يزداد الطلب على البرمجيات التي تعمل في بيئة موزعة عبر الشبكات التي تحتوي أنواعا مختلفة من الحواسيب والأنظمة. يكمن هذا التحدي في ضرورة تطوير تقنيات تتيح بناء برمجيات تستطيع أن تتماشى مع بيئات التطوير والتنفيذ المختلفة.
٢- التسليم: يستغرق تطبيق تقانات هندسة البرمجيات زمنًا طويلاً في أغلب الأحيان. ومع تسارع إيقاع العمل والتغييرات السريعة المطلوبة أصبح من الضروري تطوير تقنيات تتيح تسليم البرمجيات بزمن أقل دون التأثير على جودتها.
٣- الثقة: مع دخول البرمجيات في جميع مناحى الحياة أصبح من الضروري تطوير تقنيات تثبت للمستخدم أنه يمكنه أن يثق بالبرمجيات.

**المسؤولية المهنية والأخلاقية (١٣)**

تتطلب هندسة البرمجيات مسؤولية تتجاوز حدود تطبيق المهارات التقنية إذ يجب أن يتصرف مهندس البرمجيات بأمانة وخلق عال إذا كان يريد أن يكون مهنيًا. ونقصد بالسلوك الأخلاقى ما هو أكثر من مجرد احترام القانون؛ فما هي أهم أصول المهنه؟

١- السرية: يدخل المبرمج إلى شركة العميل ويطلع على تفاصيل عمله وأسراره المهنية لذا تقتضي أصول المهنة أن يحفظ هذه الأسرار.
٢- الكفاءة: تقتضي أن يبذل مهندس البرمجيات قصارى جهده وأن يعمل بسرية. وهذا يعنى ألا يقدم على عمل يعرف مسبقًا أنه لا يملك المهارات الكافية له.
٣- حقوق الملكية الفكرية: تقتضي أن يكون المهندس مطلع على قوانين الملكية الفكرية كبراءات الاختراع وحقوق النشر ويجب أن يضمن المهندس حماية الملكية الفكرية لعملائه.
٤- عدم إساءة استخدام الحواسيب: تقتضي ألا يستخدم مهندس البرمجيات مهاراته لإساءة استخدام حواسيب الآخرين. وقد تكون الإساءة بسيطه أو فادحة (كنشر الفيروسات).

وقد قامت مؤسستا IEEE وACM بوضع القواعد الأخلاقية لممارسة المهنة. تتضمن ثمانية بنود تتعلق بسلوك مهندس البرمجيات وبالقرارات التي عليه أن يأخذها. ويمكن الاطلاع عليها في الموقع www.acm.org/about/se-code.

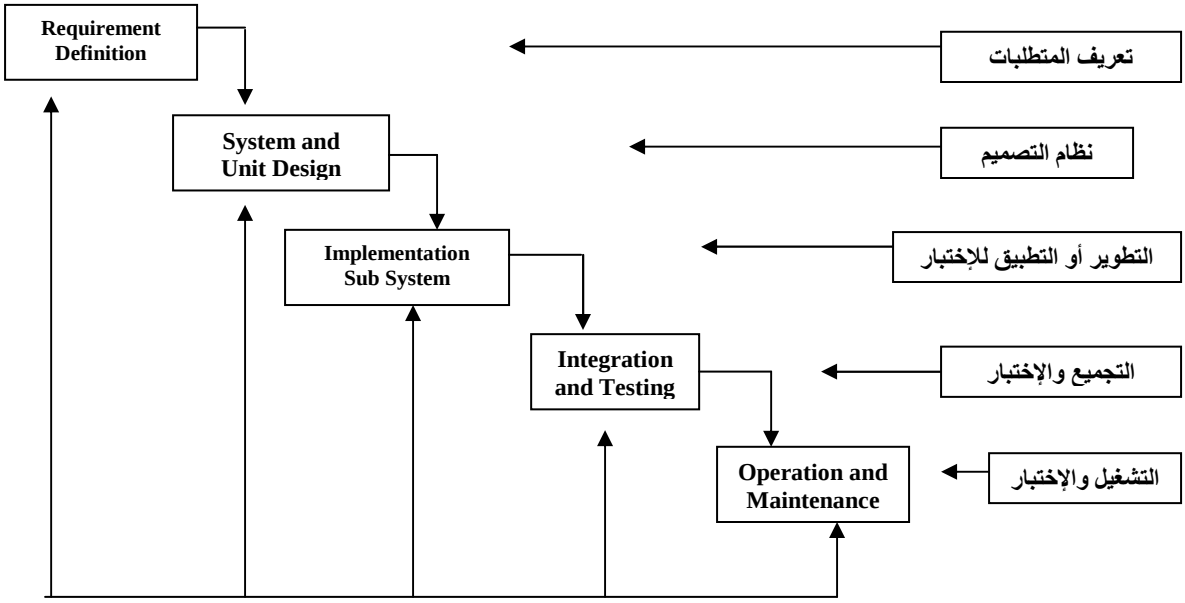
**ثانياً: مداخل تطوير إدارة عمليات البرمجيات**

يوجود العديد من نماذج تطوير النظم في عالم هندسة البرمجيات، ومع اختلاف طرائق استخدامها وتعدد تطبيقاتها قد يخلق أنواعاً مختلفة من المشاكل مما قد يؤدي إلى زيادة الجهد والكلفة أثناء تطوير هذه النظم. وهناك عدد من النماذج لإدارة عمليات البرمجيات منها:- Waterfall Model - Iteration Model -V-shaped Model- Spiral Model AgileSoftware Development- Extreme Programming (Xp) Model- Devops

وتعدهذه الطرق الأكثر انتشاراً واستخداماً في الوقت الحالي وسنقوم بعرض هذه النماذج. (١٤)

❖ **الطريقة الإنحدارية أو نموذج الشلال Waterfall model**(١٥، ١٦)

يعد نموذج الشلال من النماذج الأقدم والأبسط ويمثل إطار عمل خطى حيث تسير دورة الحياة بشكل تدريجي وواضح، وقد بدأ تصميمه عام ١٩٦٠ وانتهى ١٩٧٠، فمن المعروف عنه أنه يتجه من أعلى إلى أسفل؛ حيث ينتقل من عملية إلى أخرى دون تكرار أى عملية أو الرجوع لعملية سابقة أو تخطى عملية أخرى، ويستخدم دائمًا للمشاريع والبرامج الصغيرة، وقد كان هذا النموذج لفترة طويلة أساس عمل كثير من المؤسسات مثل وزارة الدفاع الأمريكية، واستنبط منه العديد من النماذج الأكثر تعقيداً. ويوضح الشكل (١) مراحل النموذج.




**شكل (١) مراحل نموذج Waterfall**

ويعد هذا النموذج الأكثر استخداماً في هندسة البر مجيات، وله مميزات وعيوب هي:(١٧)

- مميزات النموذج:

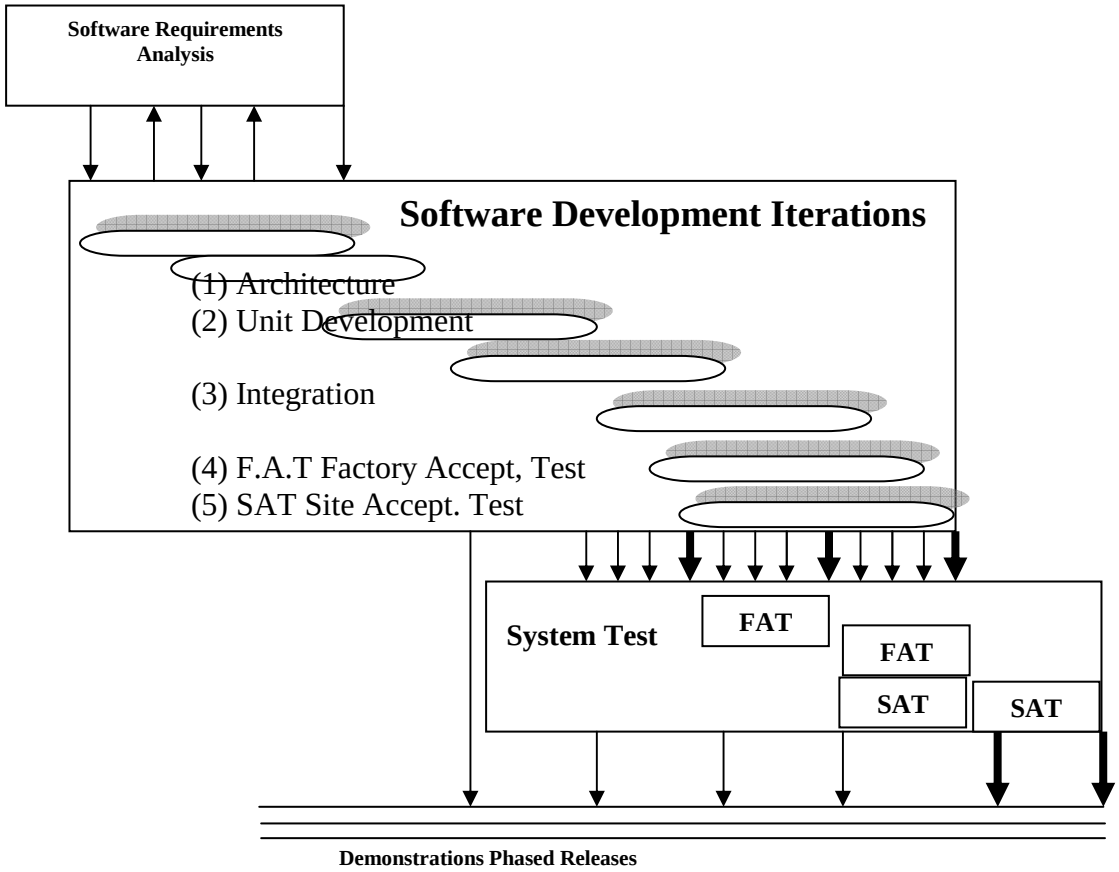
سهولة فهمه وتطبيقه - مستخدم بشكل واسع ومعروف - معامله واضحة ومحددة: يعرف التحليل قبل التصميم والتصميم قبل التنفيذ - يعمل هذا النموذج مع النظم الكبيرة.

- عيوب النموذج:

مثالي - لا يطابق الواقع بشكل جيد أحياناً - لا يعكس الطبيعة التكرارية للتطوير الاستكشافي - غير واقعي لتوقع متطلبات دقيقة في بداية المشروع - يتطلب فترة طويلة أثناء تطوير النظم - لا يدعم إدارة المخاطر - ارتفاع كلفة الصيانة التي قد تنتج بسبب تغييرات في متطلبات النظام - التكلفة المرتفعة في تطوير النظم الصغيرة.

### ❖ النموذج التكرارى Iteration Model (١٨)

يعمل النموذج فى إطار عمل متكرر، وغالباً ما تستخرج متطلبات النظام أثناء سير المشروع لهذا تتكرر العملية ويعاد العمل على المراحل المبكرة من المشروع خاصة في النظم الكبيرة، ويمكن فى هذا النموذج العودة لأي مرحلة سابقة في عملية التطوير وإعادتها عدة مرات، ويوضح الشكل (٢) مراحل النموذج.




**شكل (٢) مراحل نموذج Iteration**

- **مميزات النموذج:**

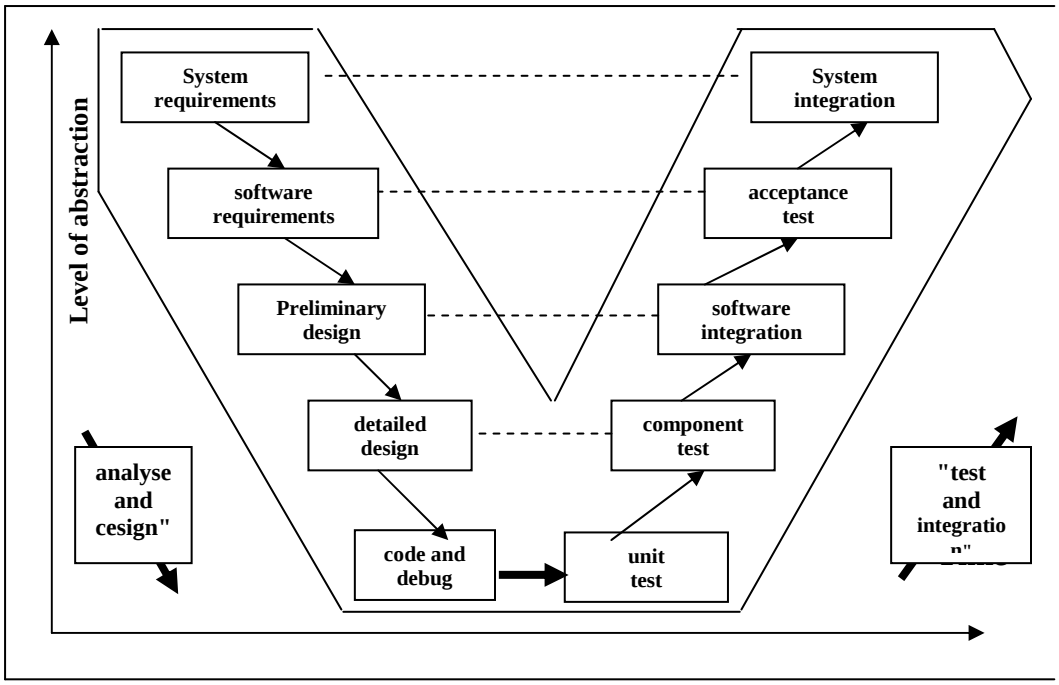
**يعطي فرصة أكبر لتفاعل المستخدم مع المشروع - يركز على مهارات التواصل والتنسيق أثناء تطوير المشروع- قدرة النموذج على التطوير من خلال استيعاب المتطلبات المتغيرة.**

- **عيوب النموذج:**

**يأخذ وقت أطول في تطوير المشروع - قديؤدى إلى بعض التشويش بعد كل مرحلة بسبب عدم انتهاء المشروع لذا يجب تطوير آلية للتحكم في الطلبات الجوهرية للمشروع.**

### ❖ V- shaped Model (19)

يعتبر هذا النموذج شبيه بنموذج Waterfall من حيث تسلسل العمليات وعدم البدء بالعملية التالية قبل الانتهاء من العملية الحالية، غير أن هذا النموذج متقدم من ناحية اختيار الأنشطة للمراحل المختلفة، حيث تركز المرحلة الأولي على معمارية النظام وتصميمه بشكل عام والآخري على تصميم مكونات النظام بشكل تفصيلي، والشكل (٣) يوضح مراحل النموذج.




**شكل (٣) مراحل نموذج V-shaped**

**ولهذا النموذج العديد من المميزات والعيوب كما يلي:** (20)

- **مميزات النموذج:**

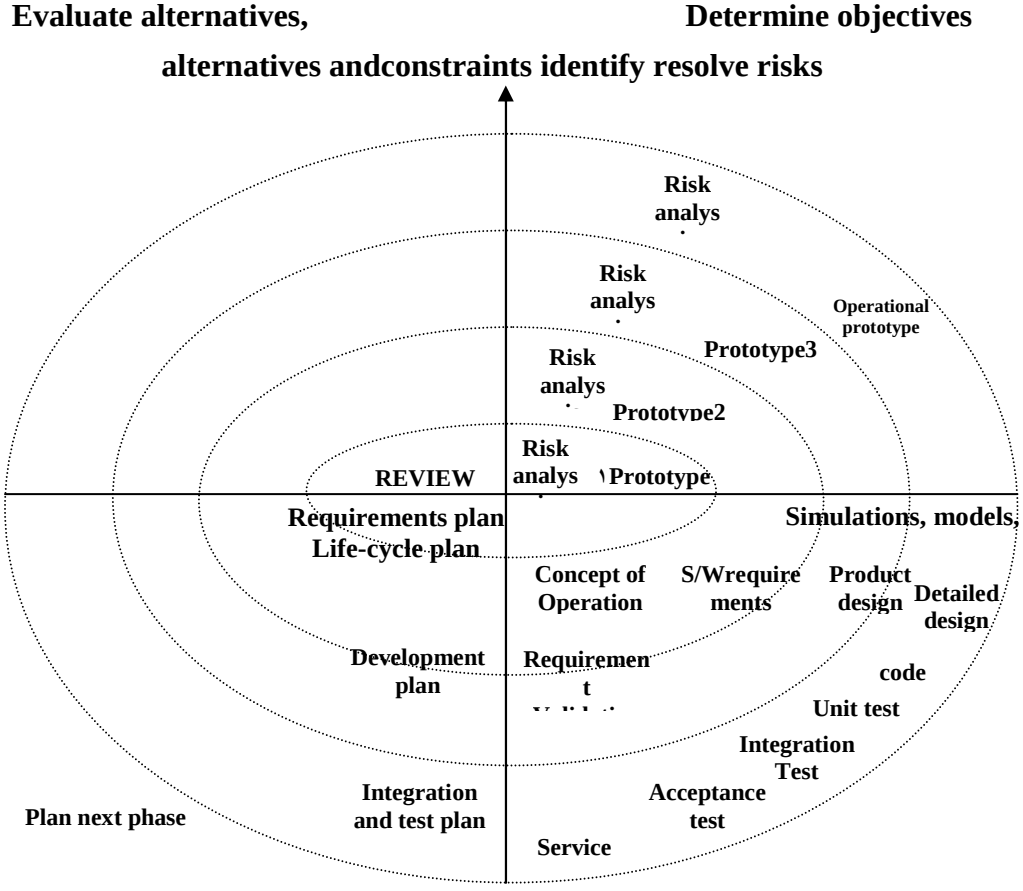
سهل وبسيط الاستخدام - كل مرحلة لها محدداتها الحاضنة - فرصة نجاح تطوير المشروع باستخدام هذا النموذج أكبر من نموذج Waterfall لاختبار الخطة في مرحلة مبكرة - يعمل جيداً مع المشاريع الصغيرة التي تكون فيها المتطلبات واضحة.

- **عيوب النموذج:**

غير مرن بشكل كافٍ مثل نموذج Waterfall - التعديل ضمن هذا النموذج صعب ومكلف - البرمجيات باستخدام هذا النموذج تنتج في وقت متأخر في مرحلة التطبيق - لا يزود النموذج بطريقة واضحة لحل المشاكل أثناء مرحلة الاختبار.

❖ **النموذج الحلزونى Spiral Model**(21)

النموذج الحلزوني الذي اقترحه "Boehm" هو نموذج تطويري لعمليات البرمجيات، ويتم فيه دمج فعالية التطوير مع إدارة المخاطر من أجل التحكم بها وتقليلها ويعتبر النموذج الحلزوني طريقة واقعية لتطوير نظم وبرمجيات واسعة النطاق. ويوضح شكل (٤) مراحل النموذج.




**شكل (٤) نموذج Spiral**

**وللنموذج الحلزوني العديد من الميزات والعيوب كما يلي:** (22)

- **مميزات النموذج:**

يعطي هذا النموذج أهمية بالغة لتحليل المخاطر - يعتبر نموذج جيد للمشاريع الكبيرة والحرجة - البر مجيات تنتج في مرحلة مبكرة من دورة حياة النظام.

- **عيوب النموذج:**

يمكن أن يعتبر هذا النموذج مكلف من ناحية تطوير النظم - عملية تحليل المخاطر ضمن هذا النموذج تحتاج إلى خبرة عالية - نجاح المشروع يعتمد بالأساس على مرحلة تحليل المخاطر - لا يستخدم مع المشاريع الصغيرة.

❖ **نموذج البرمجة الرشيقة (أجايل Agile)** (٢٣، ٢٤، ٢٥ )

بدأ عدد من محترفي تكنولوجيا المعلومات العمل على الطرق الجديدة لتطوير البر مجيات، وفي عام ٢٠٠١ في مؤتمر في ولاية يوتا، قاموا بإنشاء ما يسمي بـ Agile Manifesto، تم تطوير هذه الأساليب بناء على القاعدة نفسها التي تتمثل في أن أفضل طريقة للتحقق من أحد الأنظمة هي تقديم إصدارات العمل إلى العميل ثم تحديثها وفقا لملاحظاتها، بني مؤلفو Agile منهجياتهم على أربعة مبادئ هى:

١. الهدف الرئيسي هو تطوير برمجيات ترضي العملاء من خلال الاستمرار في تقديم برامج عاملة والحصول على تعليقات من العملاء حولها.
٢. قبول التغييرات في المتطلبات في أي مرحلة ليشعر العملاء براحة أكبر في عملية التطوير.
٣. التعاون بين المطورين والزبائن (رجال الأعمال) عي أساس يومي في جميع المراحل.
٤. التطوير على أساس الإختبارات أي كتابة إختبار قبل كتابة التعليمات البر مجية.

وتعني Agility باختصار تجريد الكثير من الثقل المرتبط بمنهجيات تطوير البر مجيات التقليدية من أجل تعزيز الاستجابة السريعة للبيئات المتغيرة، والتغييرات في متطلبات المستخدم، وتسريع مواعيد تسليم المشروع؛ حيث تكمن فلسفتهم في تقديم العديد من إصدارات العمل للبرنامج في وقت قصير، ثم تحديث البرنامج وفقا لتعليقات العملاء.

لذا فهي طريقة سريعة لتطوير البرامج، وليست كطريقة الـ Waterfall التقليدية والمرتكزة على التحليل الكامل للنظام قبل بداية تطويره، وهناك طريقتان من الممكن اتباعها عند تطوير البر مجيات الخاصة ألا وهم: نموذج الشلال المعروف باسم الـ Waterfall وطرق التطوير الرشيقة المعروفة باسم Agile.

يعتمد Agile مبدأ التسليم التكراري والمبكر للمنتج، ويرحب بتغيير المتطلبات حتى إن ظهرت في مراحل متقدمة من التطوير، ويتم وفقا لها تسليم برمجيات صالحة للإستعمال على فترات منتظمة، كما أن العميل هنا يعتبر جزءًا من فريق التطوير ويتم إشراكه في مراحل تطوير المشروع؛ حيث يمكنه إضافة المتطلبات.

حصلت agile على اهتمام متزايد بشكل كبير ضمن مجال هندسة البرمجيات، واسترعت اهتمام مهندسي البرمجيات والباحثين حول العالم، وقد لاقت منهجيات ASD القبول منذ عام ١٩٩٠ عندما فرضت بشكل scrum – crystal – extreme- programming ومنهجيات أخري، ويزعم مؤيدوها أن أهم ما يميزها البساطة والسرعة.

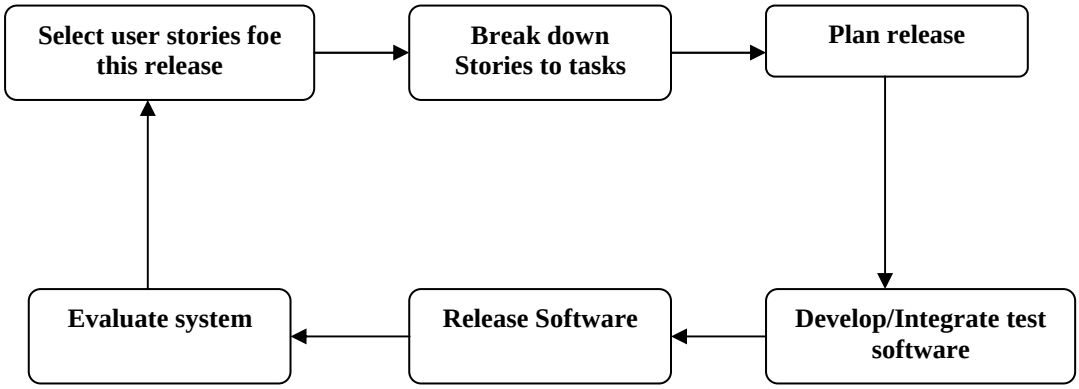
وقد نصت اللائحة التنفيذية لمحاور الـ agile في عام ٢٠٠١ على اتباع اثني عشر مبدأ لتطبيق منهجية الـ agile، وهي:

- الأولوية الأهم في العمل هي إرضاء المستفيد من خلال عمل مستمر وتسليم مبكر لأي جزء من أجزاء المشروع.
- الترحيب بأي تغيير في المتطلبات حتى وإن كان في وقت متأخر، وذلك رغبة في إضفاء سمات تنافسية ترضي العميل.
- تسليم المشروع على أجزاء تعمل فعلياً وفي فترات تتراوح من أسبوعين إلى شهر لكل دورة (Iteration) مع العمل على أقصر طريق لتسليم كامل المشروع وعدم تكرار الأجزاء.
- العمل الجماعي ما بين المبرمجين والتقنيين وأصحاب الخبرة العملية في العمل الإداري.
- بناء المشروع بواسطة مجموعة من الأفراد المحفزين من خلال توفير البيئة الجاذبة والدعم الوثقة اللازمة لإتمام المشروع.
- الحث على اللقاء الدوري والاجتماعات وجهاً لوجه نظراً لما تحمله هذه اللقاءات الفعلية من كفاءة قصوى في نقل المعلومات وفهم التعابيير.
- المعيار الحقيقي للتقدم في المشروع هو البرمجيات المنفذة والمسلمة فعلياً.
- العمل بين كافة الأطراف يجب أن يكون على نفس الوتيرة والإلتزام طول فترة المشروع لضمان انتهائه في الفترة المحددة.
- الاهتمام والانتباه لمستوي الجودة في التصميم والبرمجة للأجزاء المنتهية من البرنامج، وعدم اعتماد معايير جودة متدنية تحت تأثير الوقت أو التكلفة.
- البساطة والدقة في وصف الأجزاء المتبقية من المشروع والبعد عن التضخيم والمبالغة في كمية العمل المبتقي.
- الفريق يجب أن يجتمع في فترات منتظمة لمناقشة السبل لرفع كفاءة العمل ومناقشة أوجه القصور التي طرأت وطريقة معالجتها.

### ❖نموذج البرمجة القصوى Extreme Programming (XP) Model(٢٦)

يستخدم هذا النموذج في تطوير النظم الصغيرة التي تعتمد على كتابة البرامج بشكل مباشر دون الخوض في عمليات التحليل والتصميم المعقدة، ويهدف Xp إلى التمكن من تطوير برامج

ناجحة بالرغم من متطلبات البرامج المتغيرة باستمرار، بتقليل تكلفة التغييرات في المتطلبات من خلال وجود دورات تطوير قصيرة متعددة بدلاً من دورة طويلة، والشكل (٥) يوضح مراحل النموذج.




**شكل (٥) نموذج Extreme Programming (XP)**

وتعمل(xp) Extreme programming على تحسين جودة البرامج والاستجابة لمتطلبات العملاء المتغيرة، من خلال خمس طرق أساسية: (٢٧)

١. التخطيط الذي يقسم المشروع إلى تكرارات عند كل عملية تكرار، وتخطيط الإصدارات لإنشاء جدول زمني لإصدار إصدارات صغيرة متكررة.
٢. الإدارة التي يمنح فيها الفريق مساحة عمل مفتوحة مخصصة.
٣. البرمجة.
٤. التصميم يتم إعادة بناء التصميم كلما كان ذلك ممكنا، ولاتضاف أي وظيفة مبكرا.
٥. الإختبار الذي يجب إختبار الوحدة البرمجية فيه قبل أن يتم إصدارها.

**أدوات العمل في منهجيةXP(٢٨)**

- تعمل على تطوير بيئة العمل في مشاريع البرامج من خلال مجموعة من الأدوات الفعالة.
- تفعيل أدوات التواصل والاتصال في بيئة العمل بين المبرمجين والمستخدمين وبشكل دائم.
- البساطة في العمل والتعامل: حيث يبتعد المبرمجين على الرسومات المعقدة أو استخدام اللغة المعقدة مع المستخدم النهائي للمشروع ومحاولة وضع التصاميم في أبسط صورة.
- التغذية الراجعة والاهتمام بآراء العملاء: حيث أن ملاحظات المستخدم النهائي للمشروع تعتبر ذات أولوية لدي الفريق ومحل اهتمام ومناقشة.
- الاحترام المتبادل بين أطراف العملية التقنية: حيث تعمل العلاقات الإنسانية والنفسية بين أفراد الفريق على إيجاد بنية متماسكة للفريق وكذلك المستخدم النهائي للمشروع.

- التشجيع والتحفيز: حيث لا يقتصر هذا الدور على المدراء بل يجب على الجميع أن يعمل على رفع الجانب المعنوي للفريق للاستمرار في العمل حتى انتهاء تسليم المشروع بنجاح.
- البرمجة الذاتية: وهي طريقة يقوم فيها اثنان من المبرمجين بالجلوس على شاشة برمجة واحدة حيث يقوم الأول بكتابة النصوص البرمجية بينما الآخر يقوم بالمراقبة والتدقيق، وبالإمكان تبادل الأدوار بعد فترة وذلك للمشاركة والتقليل من احتمالية حدوث الأخطاء.
- التخطيط: وهي نشاطات يقوم بها فريق العمل في ترتيب وتوزيع المهام البرمجية المطلوبة، بحيث يراعي فيها (الصعوبة والسرعة ومدي احتياج العميل لهذه المتطلبات)، وتتم هذه العملية على لوح أبيض بملصقات طويلة لتصبح أكثر وضوحا للنقاش والمفاضلة بينها.
- مراجعة النصوص البرمجية: وتتم المراجعة عن طريق فريق من المراجعين داخل المنظمة ممن لديهم إلمام بمتطلبات المشروع ومن ثم إعطاء رأيهم أو ملاحظاتهم حول النصوص البرمجية المكتوبة ومدي اتساقها مع متطلبات المستخدم النهائي وكذلك معايير الجودة.
- اختبار الوحدات: وهي نشاطات تتم بمساعدة بعض البرامج لاختبار الكفاءة المنطقية للنصوص البرمجية سواء كانت صفوف أو طرق، حيث يتم تمرير مجموعة من التغيرات إليها ومن ثم يتم اختبار قدرتها على استرجاع أو تغيير القيم الممررة إلى النص المبرمج.
- اختبار التكامل: وهي عملية اختبار كاملة بعد انتهاء برمجة الصفوف والطرق المطلوبة لإنهاء سناريو كامل.

ولهذا النموذج العديد من المميزات والعيوب كما يلي: (٢٩)

- مميزات النموذج:

مناسب للمشاريع الصغيرة والمتوسطة - يعتمد على النموذج التكراري - يعمل على إنتاج نظام متكامل بشكل سريع - يعتمد على جودة المتطلبات.

- عيوب النموذج:

صعوبة استخدامة مع المشاريع الكبيرة التي تتطلب توثيق - يتطلب خبرة ومهارة عالية في التطبيق - قد يكون مكلف أحيانا، بسبب اعتماده على خبرات برمجية عالية - مرحلة الاختبار في هذا النموذج تعتبر صعبة وتعتمد على مهارة عالية.

**❖نموذج سكرم Scrum**(٣٠)

يعد Scrum هو عملية agile الأكثر شيوعا لتطوير المنتجات، وخاصة تطوير البرمجيات. فهو إطار لإدارة المشاريع للأغراض العامة ينطبق على أي مشروع بمواعيد نهائية صارمة مع متطلبات معقدة وعلى درجة من التفرد. وتتقدم المشاريع في Scrum عبر سلسلة من التكرارات تسمي sprints (السباقات) كل سباق هو عادة ٢- ٤ أسابيع طويلة يضم فريق Scrum النموذجي ما بين خمسة إلى تسعة أشخاص، لكن يمكن أن تصل إلى المئات بسهولة، لا يتضمن الفريق أيا من أدوار هندسة البرمجيات التقليدية (المبرمج أو المصمم أو المختبر) مالك المنتج هو صاحب المصلحة الرئيسي في

المشروع ويمثل المستخدمين والعملاء وغيرهم في هذه العملية، The Scrum Master مسئول عن التأكد من أن الفريق منتج قدر الإمكان وينقسم الفريق إلي:

١- مالك المنتج: وهو يقوم بتحديد خصائص المنتج المطلوبة، وكذلك أولوية كل خاصية في المنتج بالتعاون مع قائد الفريق، لذلك قد يكون مالك المنتج هو الزبون أو من ينوب عنه.

٢- قائد الفريق: يلعب قائد الفرقة دور كبير في اجتماعاته مع مدير المنتج لتحديد أولويات الخصائص وكذلك مع الفريق في تحديد الدورات لكل خاصية وأولويات المهام كما ويقوم بتقسيم وتوزيع المهام لفريق التطوير ويتابع عمل المجموعة وحالة سير المهام، فضلا عن مساعدة الفريق وإبقاء التركيز على المهام.

٣- فريق التطوير: وتكون مسؤوليته إنهاء تطوير كل دورة حسب الأولوية، كما يتعاون مع قائد الفريق على تحديد أولويات المهام والتخطيط، وحالة المهام التي يعمل عليها.

**إجتماعات** Scrum

- **اجتماعات Scrum اليومية** وهي عبارة عن اجتماع دوري يقيم به قائد الفريق حالة الدورة حيث يتم سؤال كل عضو من الفريق. (ما قمت بعمله بالأمس – ماذا ستقوم بعمله اليوم – هل هناك معوقات أثرت على العمل).
- **اجتماع استعراض الدورة** ويتم فيها عرض ما تم لكل دورة لصاحب المنتج حتى ولم يكن هناك شيء فعلي فيتم عرض ما تم إنجازه فعليا.
- **اجتماع مراجعة الدورة** لمعرفة الإيجابيات والسلبيات في كل دورة وكيف يمكن الاستفادة من ذلك لتطوير كل دورة.
- **متابعة سير المنتج** يعد متابعة سير المنتج أمراً هاماً جداً في تحديد المدة والمواعيد النهائية ويمكن من خلالها تحديد درجة التعقيد أو الإنجاز في كل دورة.

أهم ما يميز الـ Scrum أن الجميع فيه مسئول عن إنجاح المنتج وما يميز هذا الإطار هو الشفافية الحقيقية في أن الفريق مطلع على المهام والمنتج، كما أن التواصلية العالية لأعضاء الفريق وكذلك العميل يساعد كثيرا في تحقيق المطلوب بأسرع وقت.

في الواقع كلا المنهجين متقاربين جداً، فالإختلافات في كثير من الأحيان خفية لكنها مهمة، ويلخص الجدول التالي أهم الفروق بين منهجتي agile

جدول (١) يوضح أهم الفروق بين منهجتي agile

| XP | Scrum |
|---|---|
| - دورات متكررة من البرمجيات الفعالة(العاملة) | - دورات متكررة من البرمجيات الفعالة (العاملة |
| - تحديثات للحالة متكررة وإعادة التخطيط | - تحديثات للحالة متكررة وإعادة التخطيط |
| - تركز على فريق والتنظيم الداخلي | - تركز على العلاقة بين فريق التطوير والتنظيم |
| - فترة العمل من أسبوع إلى ستة أسابيع | - فترة العمل من أسبوعين إلى أربعة أسابيع |
| - فريق المشروع صغير، أقل من عشرين عضو | - فريق التطوير من أي حجم |
| - يطبق لمشاريع صغيرة | - يطبق لأي نوع وحجم من المشاريع |
| - إشراك العميل | - إشراك العميل وله دور وهو مالك المنتج |
| - بيئة عمل مفتوحة، العميل جزء من الفريق | - مستوي عالي جدا من التواصل والتفاعل |

**فوائد منهجيات البرمجة الرشيقة**: (٣١)

- التحكم بتغير المتطلبات
- كشف الأعطال
- زيادة الأداء
- التسليم التكراري والتزايدي
- المرونة في التصميم
- التحسينات في الجودة

**جوانب القصور تتجلي في النقاط**:

- ينصب التركيز الرئيسي على التطوير بدلاً من التصميم والمستخدم.
- مهلة اختبار عالية.
- تعدد الفرق التي تتطلب التنسيق والاتصال العالي من مدير المشاريع.
- لا يتناسب بشكل جدي مع المشروعات الكبيرة.
- قد يتم تخصيص الكثير من الوقت لأي ميزة فردية صغيرة.
- في مشروع واسع النطاق، قد تكون تكلفة لاستخدام أساليب رشيقة مرتفعة للغاية.

**DevOps**(٣٢، ٣٣)

DevOps هو مصطلح لمجموعة من المفاهيم التي قد تحركت وانتشرت بسرعة في جميع أنحاء المجتمع التقني، مثل أي مصطلح جديد، فهو مزيج من الفلسفات والممارسات والأدوات الثقافية التي تزيد من قدرة المؤسسة على تقديم التطبيقات والخدمات بسرعة عالية، تطوير وتحسين المنتجات بوتيرة أسرع من المنظمات التي تستخدم عمليات تطوير البرمجيات التقليدية وإدارة البنية التحتية،

هذه السرعة تمكن المؤسسات من تقديم خدمة أفضل لعملائها والتنافس بشكل أكثر فاعلية في السوق.

فلم تعد فرق التطوير والعمليات في إطار نموذج DevOps " مخفية "، في بعض الأحيان يتم دمج هذين الفريقين في فريق واحد حيث يعمل المهندسون عبر دورة حياة التطبيق بالكامل، من التطوير والإختبار إلى النشر إلى العمليات، وتطوير نطاق من المهارات لا تقتصر على وظيفة واحدة. وقد تصبح فرق ضمان الجودة والفرق الأمنية في بعض نماذج DevOps أكثر تكاملاً مع التطوير والعمليات وطوال دورة حياة التطبيق.

**مميزاتها**:

- السرعة
- التسليم السريع
- الوثوقية
- التوسع
- تحسين التعاون
- الأمان

فلسفة التنمية الفكرية: يتطلب الانتقال إلى DevOps تغييراً في الثقافة والعقلية في أبسط صورها، تقوم DevOps بإزالة الحواجز بين فريقين وهما التطوير والعمليات، في بعض المنظمات قد لا يكون هناك حتي فرق منفصلة للتطوير والعمليات مع DevOps يعمل الفريقان معاً لتحسين إنتاجية المطورين وموثوقية العمليات، فهم يسعون إلى التواصل بشكل متكرر، وزيادة الكفاءة، وتحسين جودة الخدمات التي يقدمونها للعملاء، فهم يأخذون ملكية كاملة لخدماتهم، وغالباً ما يتجاوزون أدوارهم المحددة بشكل تقليدي من خلال التفكير في احتياجات العميل النهائي وكيف يمكنهم المساهمة في حل هذه الاحتياجات.

**ممارسات** DevOps

هناك بعض الممارسات الأساسية التي تساعد المؤسسات على الابتكار بشكل أسرع من خلال أتمتة وتبسيط عمليات تطوير البر مجيات وإدارة البنية التحتية، يتم إنجاز معظم هذه الممارسات مع الأدوات المناسبة.

- التكامل المستمر
- التسليم المستمر
- Microservices
- البنية التحتية كقانون
- الرصد والتسجيل
- التواصل والتعاون

**ثالثاً: التصور المقترح لإدارة الجامعات المصرية فى ضوء مداخل تطوير إدارة عمليات البرمجيات**

غدا الحاسوب هام جداً في جميع نواحي الحياة خاصة المجال المعلوماتي ومجال قواعد البيانات. حيث أصبح من الصعب حصر الشركات المصنعة للبرمجيات التي تسهل الأعمال المكتبية والإدارية في الشركات الخاصة والعامة والبنوك والمنظمات الحكومية والعلمية والجامعات من خلال تسهيل العلميات التي تجري لأي بيانات متداولة بكثرة والمتكررة في تلك المنظمات بسرعة كبيرة جداً ودقة عالية، لذا أصبحت البرمجيات العنصر الجوهري في تطور النظم والمنتجات المعتمدة على الحاسوب. وانطلاقاً من الواقع الحالي الذي يشير إلى أنه قد بات لزاماعلى كل المنظمات أن تسعى إلى امتلاك نظاماً إدارياً قوياً وفعالاً يساعدها على البقاء والاستمرار، من خلال التكيف مع بيئة الأعمال الجديدة، وإفرازات العولمة، وحدة التنافسية فى عالم سريع التغير. فلابد أن تضطلع الجامعات بدورها في إحداث تغييرات كيفية في سياساتها والأساليب المتبعة في إدارتها، وأن تستفيد من مواردها البشرية في تصميم وتطبيق مستجدات الفكر الإداري لتطوير أدائها، ولما كان مدخل تطوير إدارة عمليات البرمجيات يعد من الأساليب الحديثة التي دخلت مجال الإدارة بصفة عامة وهندسة البرمجيات بصفة خاصة، تم إعداد تصور مقترح لإدارة الجامعات المصرية فى ضوء مداخل تطوير إدارة عمليات البرمجيات كالتالى:

**المنطلقات**

قامت الباحثة ببناء التصور المقترح لإدارة الجامعات المصرية في ضوء مداخل تطوير إدارة عمليات البرمجيات على عدة منطلقات حددتها فى النقاط التالية:

١. كثرة التحديات والمتغيرات المحلية والإقليمية والعالمية التي تواجهها الجامعات والتى منها العولمة، وثورة المعرفة والمعلوماتية والتنافسية، فرضت عليها ضرورة استحداث أساليب تفكير جديدة في الإدارة وبما يمكنها من التعامل مع هذه التحديات.
٢. الاستجابة للتغيير والتطوير الذي تتطلبه المرحلة الحالية والمستقبلية؛ حيث بات واضحاً أن العالم كله يتوجه في إدارته المستقبلية نحو تنظيمات ذات سمات تنظيمية إدارية جديدة.
٣. توجه العديد من الجامعات المصرية نحو العالمية ودخول الجامعات المصرية إلى السباق العالمى.
٤. المنطلق التكنولوجى نحو مجتمع علمى تكنولوجى يرتكز إلى اقتصاد المعرفة وتكنولوجيا المعلومات.
٥. يوجد اتفاق على أن تطوير الأداء بمؤسسات التعليم العالي يرتبط بمتابعة نتائج الدراسات الحديثة في مجال الفكر الإداري وتطبيقها.
٦. قد يفيد المسئولين عن التعليم الجامعي وسياساته في التعرف على الجوانب المختلفة لمدخل حديث في إدارة التعليم الجامعي.

**أهداف التصور المقترح**

١. تحسين القدرة التنافسية للجامعات المصرية والتي يتحقق معها التواجد على خريطة الجامعات المتميزة على المستوى المحلي والاقليمي والعالمي.

٢. تقديم تصور مقترح لإدارة الجامعات المصرية ومتطلبات تنفيذه

٣. تحديد عمليات إدارة الجامعات واليات تنفيذها.

## مراحل التصور

يوضح الشكل التالى مراحل التصور المقترح:


**شكل (٦) مراحل التصور المقترح (من إعداد الباحثة)**

**المرحلة الأولى: دراسة الوضع الراهن للجامعة**

**آليات التنفيذ:**

- التحليل البيئى(SWAT)

تحليل البيئة الداخلية والخارجية لتحديد احتياجات الجامعة من الموارد البشرية والمادية وتخطيط الأنشطة والمهام.

- تحديد المتطلبات اللازمة لتحقيق أهداف الجامعة وتشمل:
  - متطلبات وظيفية.
  - متطلبات غير وظيفية.
- تحديد المخاطر المحتملة

ويمكن أن تعد لائحة لأنواع المخاطر المحتملة وتوجد على الأقل ستة أنواع هى:

- مخاطر التقانة
- مخاطر فى البشر
- مخاطر تنظيمية

- مخاطر فى الأدوات
- مخاطر فى المتطلبات
- مخاطر فى التقدير

➢ التهيئة ووضع رؤية الجامعة لتطوير إدارتها فى ضوء نتائج التحليل البيئى للوضع الراهن للجامعة. ويتطلب الآتى:

- اقناع قيادات الجامعة بأهمية تطوير إدارة الجامعة وانعكاساته على الجامعة ومنسوبيها.
- نشر ثقافة وفكر التنافسية لمنسوبى الجامعة.
- تبنى قيادات الجامعة لقيم الشفافية والمسائلة والمحاسبية والتميز والإبداع والريادة والمشاركة والإلتزام وتمكين العاملين للاستثمار الجيد للطاقات الإبداعية فى عملية التطوير.
- وضع رؤية الجامعة لتطوير إدارتها.

➢ تحيد أهداف الجامعة

ويمكن إجمال الأهداف فى النقاط التالية:

- إعداد خريجين متميزين ذوى كفاءة وفعالية تعليمية ومهنية بما يلبى احتياجات ومتطلبات
- سوق العمل.
- تحقيق التميز البحثى بما يحقق الريادة العالمية للجامعة.
- تكوين صورة وسمعة متميزة للجامعة.
- تفعيل اتفاقيات التعاون والشراكة مع المؤسسات المحلية والوطنية والدولية.
- الارتقاء بالأداء الإدارى والتقنى والمعلوماتى للجامعة.
- تحسين كفاءة ورضا الكفاءات البشرية المتميزة والمحافظة عليها.

**المرحلة الثانية: التخطيط لإدارة الجامعة**

**آليات التنفيذ**

➢ تحديد محاور الخطة وصياغة الأهداف والأنشطة والوزن النسبى لكل محور من المحاور وضعت الباحثة عددا من المحاور كالتالى:

- التعليم والتعلم ٢٠٪
- البحث العلمى ٢٠٪
- الموارد البشرية ٢٠٪
- الموارد المادية ١٥٪
- التقنية ونظم المعلومات ١٥٪
- الشراكات ١٠٪

➢ تحديد المدى الزمنى لتنفيذ الأنشطة.
➢ توزيع المهام (مسئول التنفيذ – مسئول المتابعة).
➢ تحديد الموارد المادية اللازمة لنتفيذ الأنشطة.
➢ إدارة المخاطر

تتألف إدارة المخاطر من المراحل التالية:

- تحديد المخاطر: وفيها يتم إعداد قائمة بالمخاطر المتوقعة.
- تحليل المخاطر: تقدير احتمالية وقوع هذه المخاطر ونتائجها.
- التخطيط لمواجهة المخاطر: توضع خطة للتعامل مع المخاطر.
- مراقبة المخاطر: تقدر باستمرار وتراجع خطط مواجهة المخاطر كلما توافرت معلومات حولها.

➢ إدارة التغيير

يمكن أن نذكر أسباب التغيير منها:

- ظهور متطلبات جديدة.
- تغيير بيئة العمل.
- ظهور أخطاءتحتاج إلى تصحيح.
- إضافة تجهيزات جديدة على الجامعة.
- ضرورة تحسين الأداء الإدارى بالجامعة.

ويمكن تحديد مرحل إدارة التغيير كالآتى:

١. تحليل التغيير المطلوب.
٢. اقتراح التغيير.
٣. التخطيط للتغيير
٤. تنفيذ التغيير (نظام جديد)

➢ تحديد متطلبات تنفيذ الخطة

- توفير البنية التحتية المناسبة لتنفيذ خطة تطوير إدارة الجامعة وأنشطتها
- بناء ثقافة تنظيمية تشجع على نقل وتبادل الخبرات المختلفة، وتؤكد على الإبداع والتميز.
- تحديث الهيكل التنظيمي للجامعة بشكل مستمر مما يمكن من تطوير القدرة التنافسية بها.
- توسيع فرص مشاركة رأس المال البشري بالجامعة في صنع واتخاذ القرارات.
- وضع السياسات العامة اللازمة لتعظيم الاستفادة من تطوير القدرة التنافسية، كذلك اتباع سياسة المقارنة المرجعية مع إحدى الجامعات المتميزة إداريا على المستوى العالمي.
- تأسيس نظام لتفعيل قنوات الاتصال بين الجامعة وقطاعات المجتمع المحلي بما يحقق مشاركتهم في أنشطة الجامعة دون الإضرار بحريتها الأكاديمية
- وضع معايير موضوعية لتقييم أداء القوى البشرية بالجامعة بما يتناسب مع اسس ومبادئ ومتطلبات تطوير الإدارة الجامعية.

**المرحلة الثالثة: التنظيم**

**آليات التنفيذ**

- تهيئة البيئة الخارجية

١- الحكومة: دعم الحكومة لتنافسية الجامعات يأتي عبر ثلاث قنوات التشريع والتنظيم والتمويل.

- التشريع يتعلق بتنظيم الجامعات، عبر التحول من ميزانيات الأداء إلى الميزانيات التي تركز على الأداء والكفاءة، ومنح التمويل بناء على نتائج الأداء والعمل المتفوق.
- التنظيم عبر منح الجامعات الاستقلالية التنظيمية والإدارية والمالية وإدارتها بما يتضمن مرونة القرار.
- التمويل المستمر والدائم حتى يمكن للجامعات القيام بأدوارها بكفاءة، وبما يضمن قدرتها على التحول نحو مجتمع المعرفة والاقتصاد المعرفي.

٢- المجتمع: للمجتمع توقعات معينة من الجامعات تتمثل في تعليم وتدريب وتأهيل المواطنين واستيعابهم في هذه المؤسسات، وهذا يتطلب تفهما مجتمعيا داعما لتعزيز تنافسية هذه المؤسسات بما يتوافق مع المعايير الدولية المعتمدة.

٣- مؤسسات ضمان الجودة والاعتماد والتنافسية: حتى يمكن للجامعات العمل وفق شروط وبيئة تنافسية صحية، ووفق معايير تقويم وجودة متسقة مع المعايير الدولية.

- تهيئة البيئة الداخلية:

الثقافة التنظيمية: المبنية على قيم التميز والابداع والابتكار، والمبادرة والتمكين الادارى.

١- القيادة الجامعية: القادرة على تبنى رؤية استراتيجية تسمح بالتحول نحو الاقتصاد المعرفى، والقدرة على حفز منسوبى المؤسسات والتأثير فيهم وخلق فرق العمل، والانتماء والولاء للمؤسسة.

٢- الموارد والكفاءات: يعتبر الانسان هو المحرك الحقيقى لأي تنظيم ومؤسسات التعليم مؤسات معرفية بدرجة كبير تحتاج من أجل العمل بكفاءة وفاعلية أن تضم بين جنباتها كفاءات ذات مؤهلات وقدرات متميزة، هذه الكفاءات والقدات هى من يخلق التميز والفارق بين الجامعات خاصة عندما يتم دعمها بموارد إالية ومادية وتقنية تسهل عملها وتساعدها على الابداع والابتكار.

٣- البنية التحتية: بنية الجامعات تمثل البيئة التى تخص عمليات وأنشطة المؤسسة، وتوفر البيئة المناسبة (من مبانى ومعامل، مختبرات ومصادر معرفة.. الخ) يدعم أداء تلك العمليات والأنشطة ويوفر تعزيزا مهما فى الانصراف نحو الابداع والابتكار بدلا من البحث عن متطلبات العمل الأساسية.

- التكامل حيث يتم الربط بين المكونات المختلفة للنظام بحيث تدمج لتصبح واحدة تشكل على إثرها النظام، ويراعي في هذه المرحلة تكاملية الأجزاء المختلفة مع بعضها البعض من دون التأثير السلبي على بقية أجزاء مكونات النظام.

- الاختبار تبدأ عملية الاختبار مع أول مرحلة لأي نظام وتشمل اختبار الخطة واختبار جمع المتطلبات وكذلك التصميم والتنفيذ والتكاملية.

**المرحلة الرابعة: تنفيذ الخطة ومتابعتها**

**آليات التنفيذ**

- وضع المحاور والأهداف والأنشطة لكل محور فى صورة بطاقات باستخدام بطاقة الأداء المتوازن كالآتى: (نماذج للمحاور باستخدام بطاقة الأداء المتوازن)

**جدول (٢) لمحور التقنية ونظم المعلومات**

| الأهداف | الوزن ٪١٠٠ | الهدف | المبادرات أو الأنشطة | المؤشرات | التكلفة | | المنجز | طريقة القياس | النتيجة النهائية * | المستهدف |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | تمويل حكومي | تمويل ذاتي | | | | |
| الإرتقاء بالأداء الإدارى والتقنى والمعلوماتى للجامعة | ٪١٥ | ٪٩٠ | ـ التحول نحو تطبيق الادارة الإلكترونية<br>– الاستثمار في البنية التحتية لتقنية المعلومات بما يدعم عملية التدريس والبحث العلمي وانتاج المعرفة<br>– توفير وتحديث وصيانة مصادر المعرفة وأوعيتها<br>– الاستثمار في نقل وتوطين التقنية بما يعزز الاقتصاد المعرفي<br>ـ متابعة مستجدات التقنية وتوفير متطلبات العملية التعليمية والبحثية. | | | | ٦٥ | كمية، رقمية، نسبة مئوية | ٧,٢ | ١,٨ |
| إجمالي المحور | ٪١٥ | ٪٩٠ | | | | | | | | |

- تحديد معوقات التنفيذ وكيفية التغلب عليها
  - ضعف البنية التحتية بالجامعات.
  - جمود القوانين واللوائح التي تضعف من نجاح الجامعة.
  - ضعف المناخ التنظيمي بالجامعة.
  - ضعف قناعة القيادات بالعائد الذي يتحقق من تطويرالإدارة
  - ضعف قنوات التواصل بين الجامعة والقطاعات المستفيدة.
  - شكلية ونمطية البرامج التدريبية للموارد البشرية بالجامعة.
- آلية متابعة تنفيذ الخطة

*يتم حساب النتيجة النهائية على أساس أنها تساوي وزن القياس مضروب فى المنجز فعلا /الهدف

## المراجع:


١- محمد سعيد عبدالمجيد (٢٠٠٦): "**قانون تنظيم الجامعات وجودة التعليم" دراسة ميدانية**، المؤتمر الدولي الثاني لقسم علم النفس- سلوك الإنسان وتحديات العصر، ١٨- ٢٠٠٦/٤/٢٠م، جمهورية مصر العربية، جامعة المنيا، كليةالآداب.

**2- Natalija Sedziuviene, Jolita Vveinhardt (2009): The Paradigm of Knowledge Management in Higher Educational Institutions**
**http://internet.ktu.lt/lt/mokslas/zurnalai/inzeko/65/1392-2758-2009-5-65-079.pdf**

٣- **غسان العمري (٢٠٠٤): الاستخدام المشترك لتكنولوجيا المعلومات وإدارة المعرفة لتحقيق قيمة عالية لأعمال البنوك التجارية الأردنية، أطروحة دكتوراه غير منشورة، جامعة عمان العربية للدراسات العليا، عمان، الأردن.**

**4-** Janzon, T. (2003)**The Knowledge in Innovation: A Study of SKF Nova**, Master Thesis, Gothenburg University.

5- Kang, T. (2003) The Knowledge Advantage**: Tracing and Testing the Impact of Knowldege Characteristic and Relationship on Project Performance,** PHD. Digital Dissertation University of Califotnia, Los Angeles, wwwlib, umi.com/Disseration/Seach

٦- غيداء ربداوى (٢٠١٨)**: هندسة البرمجيات**، سوريا، الجامعة الافتراضية السورية.

٧- 2016**Softwere Development Life Cycle(SDLC) متاح على الإنترنت.**

٨- غيداء ربداوى (٢٠١٨): مرجع سابق

٩- غيداء ربداوى (٢٠١٨): مرجع سابق

**10-** QA &Test Conference(2019): **SoftwereTesting & QA,** Spain.

١١- إيمان عيسي، تالة حجيج، يزن القطشة (٢٠١٨): **طرق تطوير البرمجيات وإدارة الفرق البرمجية**، كلية الهندسة المعلوماتية، جامعة دمشق.

**١٢-** عبد الحميد بسيوني (٢٠٠٥): **أساسيات هندسة البرمجيات**، دار الكتب العلمية للنشر والتوزيع.

١٣- غيداء ربداوى (٢٠١٨): مرجع سابق

١٤- نبيل محمد على منصر، محمد عبد الله الهاجرى (٢٠٠٩): **نموذج مقترح فى إدارة عمليات البرمجيات**، مجلة العلوم والتكنولوجيا، المجلد (١٤)، العدد (١).

١٥- زاهر الحاج حسين:)2006(**هندسة البرمجيات ثنائية الهندسة والإدارة**، الطبعة الأولي، دار شعاع للنشر والعلوم.

**16-** Ian Sommerville(2004)**:"Software Engineering "** , Addison Wesley, 7th edition.

**17-** Karlm (2006): **" software Lifecycle models"** , KTH**.**

**18-** **Lifecycle models,** National Instruments Corporation, 2006.

**19-** Steve Easterbrook (2001)**:Software Lifecycles, University of Toronto of** Computer Science**.**

**20-** Rlewallen,( 2005): **" Software Development Life Cycle Models.**

**21-** Ian Sommerville(2004)**:"Software Engineering** ", Addison Wesley, 7th edition.

**22-** Barry Boehm edited by Wilfred J. Nansen (2000): **" Spiral Development: Experience, Principles, and Refinements".**

**23-** Bharat Choudhary and Shanu K Rakesh. (2016): **An Approach using Agile Methods for Software Development.** Bilaspur, Chhattisgarh ,India**.**

**24-** Gauray kumar, Pradeep kumar Bhatia(2012): **Impact of Agile Methodology on Software Development Process**, UCTEE**.**

25- Malek AIzewairi, Mariam Biltawi, Weel ETaiwan, Adnan shacut( 2017): **Agile Software Development Methodologies Survey of Surveys**, The ECE Department, The University of Michigan- Dearbon, Dearbon, US.

**26-** Karlm (2006): **" software Lifecycle models"** , KTH**.**

٢٧- إيمان عيسى، تالة حجيج، يزن القطشة (٢٠١٨): مرجع سابق.

٢٨- إيمان عيسى، تالة حجيج، يزن القطشة (٢٠١٨): مرجع سابق.

**٢٩-** Karlm (2006): **" software Lifecycle models"** , KTH**.**

٣٠- إيمان عيسى، تالة حجيج، يزن القطشة (٢٠١٨): مرجع سابق.

٣١- إيمان عيسى، تالة حجيج، يزن القطشة (٢٠١٨): مرجع سابق.

**32-** F.M.A. Erich, C. Amint & M. Daneva (2017): **A Qualitative Study of DevOps usage in Practice**, Journal of Software: EVOLUTION AND PROCESS**.**

**33-** Mojtaba Shahin, (2015): **Arvhitecting for DevOps and Continucus Deployment.** The University of Adelaide. Australia.

***34-*** www.acm.org/about/se-code.

## *Conceived proposal for the management of Egyptian universities in the light of approaches to the development of software operations management.*

***Abstract***

Many new approaches and trends have emerged in management and many of them have been applied in factories and companies for many years, but they have not been talked about or applied in the management of educational institutions only in the past few years.Based on the challenges of the century at the global and regional levels, which led to rapid and radical changes in all aspects of life, several pivotal aspects in the institutions of higher education, in addition to the new information and communication concepts brought about by the revolution. As well as the challenges faced by Egyptian universities, which impose on them the need to develop new thinking methods in management in general and university management in particular; Strengthen the links between them and society, as well as achieve positive interaction between them and the changes and transformations taking place around them.

In light of the importance of software development (management of software operations), and based on the researcher's interest in the need to develop performance in higher education institutions, the research has sought to identify the conceptual framework of software engineering and its importance and constraints, and development (management of software operations) and its development in contemporary management thought, and to identify the most important challenges Facing software engineering, with a presentation of some common models in software development (software operations management), and in order to provide a proposed vision for the management of Egyptian universities in the light of the approaches to the development of software operations management, based on the descriptive approach to identify the CSS Theory of Software Engineering,illustrating the most important approaches to the development of software operations management To the proposed scenario.