ماجيستير خاص
برمجة ألعاب الفيديو

tech الجامعة التيكنولوجية

الجامعة التيكنولوجية

ماجيستير خاص
برمجة ألعاب الفيدي

- طريقة التدريس: **أونلاين**
- مدة الدراسة: **12 أشهر**
- المؤهل العلمي من: **TECH الجامعة التكنولوجية**
- مواعيد الدراسة: **وفقًا لوتيرتك الخاصّة**
- الامتحانات: **أونلاين**

رابط الدخول إلى الموقع الإلكتروني: www.techtitute.com/ae/videogames/professional-master-degree/master-video-game-programming

الفهرس

01
المقدمة
صفحة 4

02
الأهداف
صفحة 8

03
الكفاءات
صفحة 14

04
الهيكل والمحتوى
صفحة 18

05
المنهجية
صفحة 32

06
المؤهل العلمي
صفحة 40

01
المقدمة

من أهم وأدق المهام عند تنفيذ مشروع لعبة فيديو هي البرمجة. البرمجة هي جوهر لعبة الفيديو، فهي العملية التي تُنشئ تعليماتها الأساسية وتُملي عملها بشكل عام. هذا يعني أنه بدون البرمجة التي صنعها المطورون، لا يمكن أن تبرز المرئيات والقصة وطريقة اللعب في عمل سمعي بصري من هذا النوع. بالتالي، فإن هذا المؤهل العلمي يقدم لطلابه كل المعرفة ليصبحوا أفضل المبرمجين في هذا المجال، بحيث ترغب أفضل الشركات في الاعتماد عليهم لتطوير مشاريعها.

تعلم كيفية برمجة أفضل ألعاب الفيديو في العالم بفضل درجة الماجستير الخاص هذه"

شهدت صناعة ألعاب الفيديو توسعاً كبيراً في السنوات الأخيرة. مع ازدياد شعبية هذا النوع من الترفيه، اضطرت الشركات العاملة في هذا القطاع إلى تصميم الألعاب ونشرها بشكل أكثر تواتراً. بالإضافة إلى ذلك، كانت هناك حاجة أيضًا إلى المزيد من الإبداع مع تزايد طلب اللاعبين على ألعاب أكثر تنوعًا ومن أنواع مختلفة وتقدم تجارب جديدة.

لهذا السبب، تطلب الصناعة من المتخصصين في برمجة ألعاب الفيديو أن يتولوا المهمة الأساسية لإنشاء التعليمات البرمجية لأعمالهم الجديدة. هذا العمل حساس ويتطلب قدراً كبيراً من التخصص، لذا يُنصح بأن تكون قد خضعت لعملية تعلم شاملة ومثالية لتصبح خبيراً حقيقياً.

بالتالي، فإن درجة الماجستير الخاص في برمجة ألعاب الفيديو هي ما يحتاجه المحترفون ليكونوا قادرين على الوصول إلى شركة كبيرة في هذا المجال من خلال العمل في قسم التطوير الخاص بها. خلال هذا المؤهل العلمي، سيتعلم الطلاب أساسيات البرمجة وهندسة البرمجيات، وبنية البيانات والخوارزميات، والبرمجة الموجهة للكائنات وغيرها من القضايا المحددة مثل محركات ألعاب الفيديو أو البرمجة في الوقت الحقيقي.

هذا يضمن حصول الطلاب على أفضل معرفة ممكنة حتى يتمكنوا من تطبيقها مباشرةً في مجالات عملهم.

تحتوي **درجة الماجستير الخاص في برمجة ألعاب فيديو** على البرنامج التعليمي الأكثر اكتمالا وحداثة في السوق. أبرز خصائصه هي:

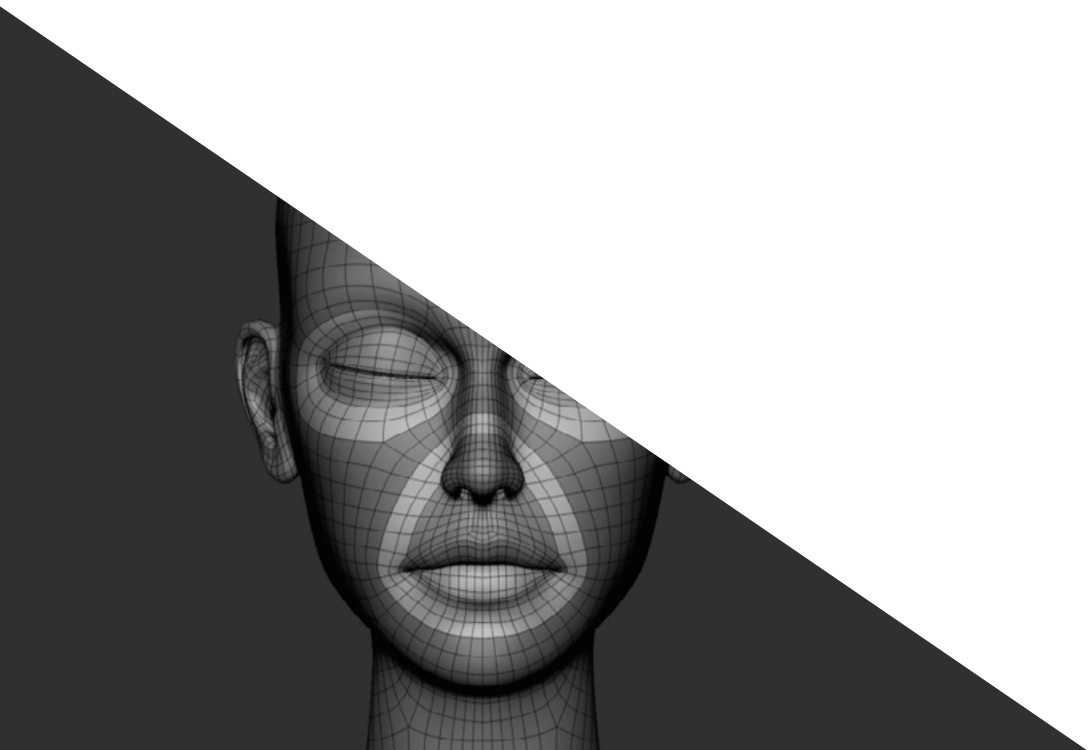
- تطوير الحالات العملية المقدمة من قبل خبراء في تصميم المنتج
- محتوياتها البيانية والتخطيطية والعملية البارزة التي يتم تصورها بها تجمع المعلومات العلمية والرعاية العملي حول تلك التخصصات الأساسية للممارسة المهنية
- التمارين العملية حيث يمكن إجراء عملية التقييم الذاتي لتحسين التعلم
- تركيزها على المنهجيات المبتكرة
- كل هذا سيتم استكماله بدروس نظرية وأسئلة للخبراء ومنتديات مناقشة حول القضايا المثيرة للجدل وأعمال التفكير الفردية
- توفر المحتوى من أي جهاز ثابت أو محمول متصل بالإنترنت

> سوف ترغب أفضل الشركات في هذا القطاع في الاعتماد عليك"

لا تنتظر أكثر قم ببرمجة ألعاب الفيديو
مثل أفضل الخبراء.

قم ببرمجة ألعاب الفيديو التي تحلم بها
بفضل درجة الماجستير الخاص هذه.

أنت تريد تطوير أفضل ألعاب الفيديو في
العالم وهذا المؤهل العلمي يعلمك
كيفية القيام بذلك"



البرنامج يضم أعضاء هيئة تدريس محترفين يصبون في هذا التدريب خبرة عملهم، بالإضافة إلى متخصصين معترف بهم من الشركات الرائدة والجامعات المرموقة.

سيتيح محتوى البرنامج المتعدد الوسائط، والذي صيغ بأحدث التقنيات التعليمية، للمهني التعلم السياقي والموقعي، أي في بيئة محاكاة توفر تدريبا غامرا مبرمجا للتدريب في حالات حقيقية.

يركز تصميم هذا البرنامج على التعلّم القائم على حل المشكلات، والذي يجب على المهني من خلاله محاولة حل مختلف مواقف الممارسة المهنية التي تنشأ على مدار العام الدراسى. للقيام بذلك، سيحصل على مساعدة من نظام فيديو تفاعلي مبتكر من قبل خبراء مشهورين.

02

الأهداف

الهدف الرئيسي من درجة الماجستير الخاص في برمجة ألعاب الفيديو هو تزويد الطلاب بأفضل المعارف حتى يصبحوا أفضل الخبراء في تطوير ألعاب الفيديو في بيئتهم. لهذا الغرض، يقدم لهم هذا المؤهل العلمي سلسلة من الأدوات المطبقة في هذا المجال والتي من شأنها تحسين عملهم كمطورين وقيادتهم لتحقيق جميع أهدافهم المهنية، ليكونوا قادرين على برمجة أفضل ألعاب الفيديو في العالم.

حقق جميع أهدافك بفضل هذا المؤهل العلمي"

الأهداف العامة

- التعرف على طرق البرمجة المختلفة المطبقة على ألعاب الفيديو
- التعمق في عملية إنتاج لعبة الفيديو وتكامل البرمجة في هذه المراحل
- تعلم أساسيات تصميم ألعاب الفيديو والمعرفة النظرية التي يجب أن يعرفها مصمم ألعاب الفيديو
- إتقان لغات البرمجة الأساسية المستخدمة في ألعاب الفيديو
- تطبيق معارف هندسة البرمجيات والبرمجة المتخصصة على ألعاب الفيديو
- التعرف على دور البرمجة في تطوير ألعاب الفيديو
- التعرف على وحدات التحكم والمنصات الإفتراضية المختلفة الموجودة
- تطوير ألعاب الويب وألعاب الفيديو متعددة اللاعبين

الأهداف المحددة

الوحدة 1. أساسيات البرمجة

- فهم البنية الأساسية للكمبيوتر والبرمجيات ولغات البرمجة ذات الأغراض العامة
- تحليل العناصر الأساسية لمعلومات الحاسوب مثل الأنواع المختلفة من البيانات، والمشغلين، والتعبيرات، والبيانات، وبيانات الإدخال والإخراج والتحكم
- تعلم كيفية تصميم وتفسير الخوارزميات، والتي تعتبر الأساس الضروري للتمكن من تطوير معلومات الحاسوب

الوحدة 2. بنية البيانات والخوارزميات

- التعرف على استراتيجيات تصميم الخوارزميات الرئيسية، بالإضافة إلى الطرق والمقاييس المختلفة لحسابها
- تمييز عمل الخوارزميات واستراتيجيتها وأمثلة على استخدامها في المشاكل الرئيسية المعروفة
- فهم أسلوب Backtracking واستخداماته الرئيسية

الوحدة 3. البرمجة الموجهة للكائنات

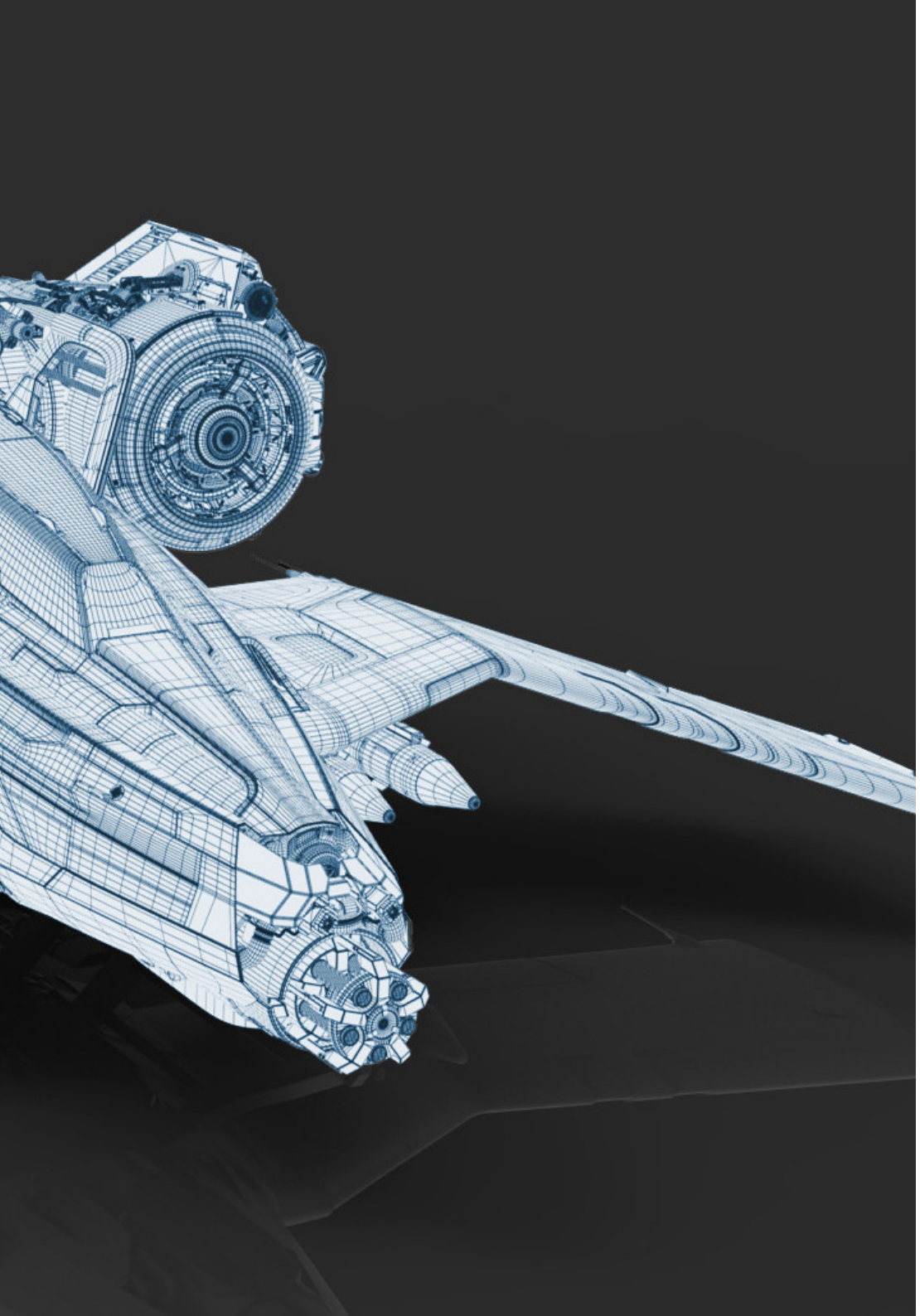
- التعرف على أنماط التصميم المختلفة للمشاكل الموجهة للكائنات
- فهم أهمية التوثيق والاختبار في تطوير البرمجيات
- إدارة استخدام سلاسل الرسائل والتزامن، بالإضافة إلى حل المشاكل الشائعة في البرمجة المتزامنة

الوحدة 4. وحدات تجكم وأجهزة لألعاب الفيديو

- معرفة الوظيفة الأساسية لملحقات الإدخال والإخراج
- فهم التداعيات الأساسية للتصميم من منصات مختلفة
- دراسة هيكل وتنظيم وتشغيل وترابط الأجهزة والأنظمة
- فهم وظيفة نظام التشغيل وأدوات التطور للأجهزة النقالة ومنصات ألعاب الفيديو

الوحدة 5. هندسة البرمجيات

- التمييز بين أساسيات هندسة البرمجيات، بالإضافة إلى عملية البرمجيات والنماذج المختلفة لتطوير البرمجيات بما في ذلك التقنيات الرشيقة
- التعرف على هندسة المتطلبات وتطويرها وصياغتها والتفاوض بشأنها والتحقق من صحتها من أجل فهم المعايير الرئيسية المتعلقة بجودة البرمجيات وإدارة المشاريع



الوحدة 6. محركات لالعاب الفيديو

- اكتشاف الوظيفة والهندسة المعمارية لمحركات لألعاب الفيديو
- فهم الخصائص الأساسية لحركات ألعاب الفيديو
- برمجة التطبيقات بشكل صحيح وفعال لمحركات ألعاب الفيديو
- اختيار نموذج ولغات البرمجة الأكثر ملائمة لبرمجة التطبيقات المطبقة لمحركات ألعاب الفيديو

الوحدة 7. أنظمة ذكية

- إنشاء المفاهيم المتعلقة بنظرية الوكيل وهندسة الوكيل وعملية الاستدلال الخاصة به
- استيعاب النظرية والتطبيق وراء مفاهيم المعلومات والمعرفة، فضلا عن الطرق المختلفة لتمثيل المعرفة
- فهم عمل المفكرين الدلالي والأنظمة القائمة على المعرفة والأنظمة الخبيرة

الوحدة 8. البرمجة في الوقت الفعلى

- تحليل الخصائص الرئيسية للغة البرمجة في الوقت الفعلى التي تميزها عن لغة البرمجة التقليدية
- التعرف على المفاهيم الأساسية للأنظمة الحاسوبية
- اكتساب القدرة على تطبيق أسس وتقنيات البرمجة الرئيسية في الوقت الفعلى

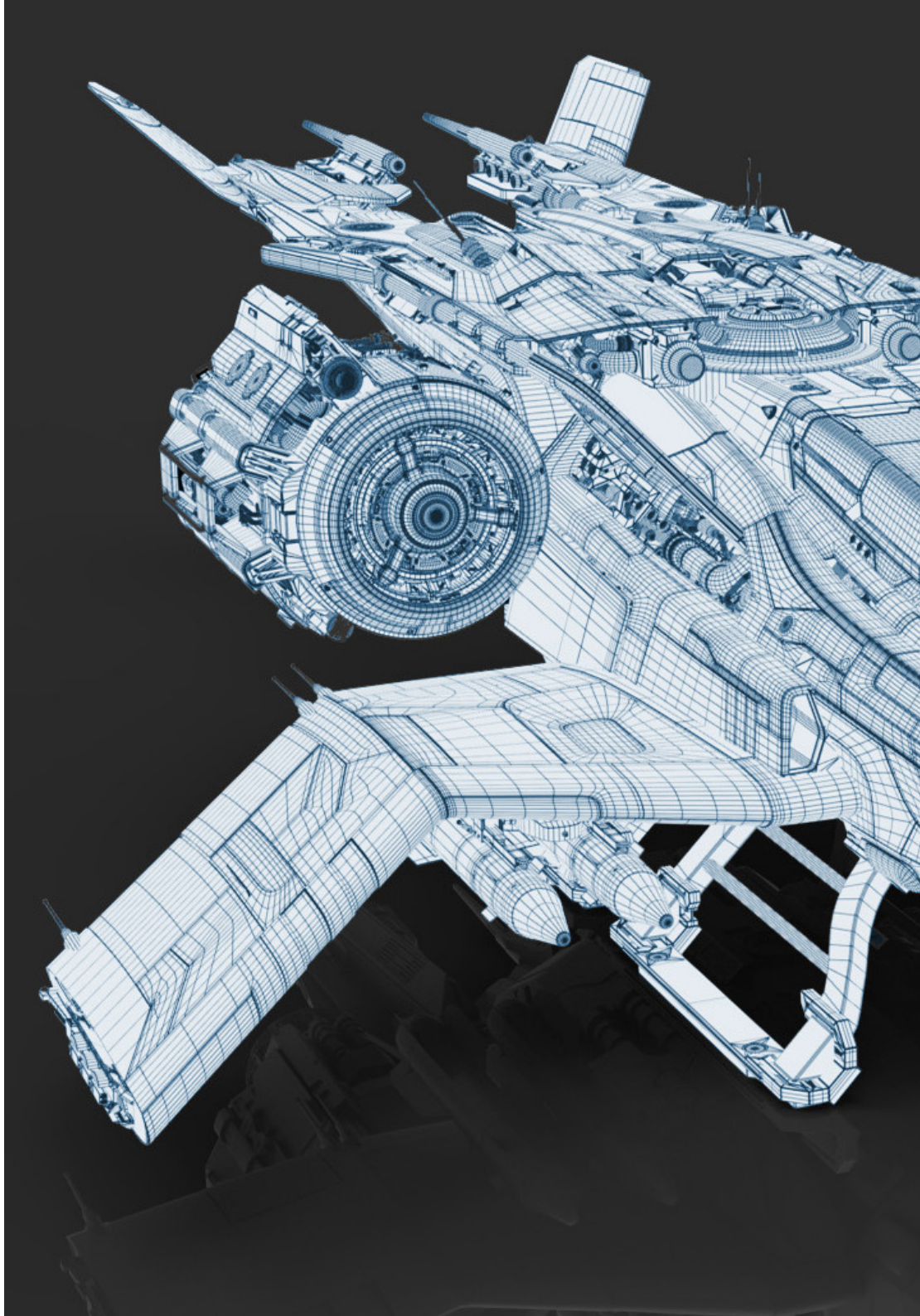
الوحدة 9. تصميم وتطوير ألعاب الويب

- تصميم الألعاب وتطبيقات الويب التفاعلية مع التوثيق المقابل لها
- تقييم السمات الرئيسية للألعاب وتطبيقات الويب التفاعلية للتواصل بشكل احترافي وصحيح

الوحدة 10. الشبكات والأنظمة متعددة اللاعبين

- وصف بنية بروتوكول التحكم في الإرسال/بروتوكول الإنترنت (TCP/IP) والتشغيل الأساسي للشبكات اللاسلكية
- تحليل الأمن المطبق على ألعاب الفيديو
- اكتساب القدرة على تطوير ألعاب متعددة اللاعبين عبر الإنترنت

كن خبيراً تقنياً حقيقياً في النمذجة
ثلاثية الأبعاد على الأسطح الصلبة"



03

الكفاءات

سيحصل طلاب درجة الماجستير الخاص هذه على سلسلة من المهارات التي ستحولهم إلى خبراء حقيقيين في تطوير ألعاب الفيديو، بحيث يمكنهم الانضمام إلى أي نوع من المشاريع في هذا المجال. بهذه الطريقة، سيتقن الطلاب القضايا المتعلقة بلغات البرمجة المحددة المختلفة المستخدمة في هذا النوع من المنتجات السمعية البصرية، بالإضافة إلى المهارات المستعرضة التي يحتاجون إلى معرفتها مثل مجال وحدات التحكم والمنصات ومحركات ألعاب الفيديو.

حقق جميع أهدافك بفضل هذا المؤهل العلمي"

الكفاءات العامة

- تصميم جميع مراحل لعبة الفيديو، من الفكرة الأولية إلى الإطلاق النهائي
- متخصص في برمجة ألعاب الفيديو
- الخوض في جميع أجزاء التطوير، بدءًا من البنية الأولية، وبرمجة شخصية اللاعب وجميع العناصر المشاركة في عملية اللعبة
- الحصول على رؤية للمشروع بأكمله والقدرة على تقديم حلول للمشاكل والتحديات المختلفة التي تنشأ في تصميم لعبة الفيديو

"أتقن جميع أنواع لغات البرمجة المطبقة على ألعاب الفيديو مع درجة الماجستير الخاص هذه

الكفاءات المحددة

- معرفة البرامج اللازمة لتكون مطور ألعاب فيديو محترف
- فهم تجربة اللاعب ومعرفة كيفية تحليل طريقة اللعب في لعبة الفيديو
- فهم الإجراء النظري والعملي الكامل لعملية برمجة ألعاب الفيديو
- أتقن لغات البرمجة الأكثر فائدة لعالم ألعاب الفيديو
- دمج البرمجة المكتسبة مع أنواع مختلفة من وحدات التحكم والمنصات
- برمجة ألعاب الويب وألعاب الفيديو متعددة اللاعبين
- استيعاب مفهوم محرك ألعاب الفيديو لكي تتمكن من البرمجة بشكل صحيح
- تطبيق المعرفة بهندسة البرمجيات على برمجة ألعاب الفيديو

04

الهيكل والمحتوى

تقدم درجة الماجستير الخاص في برمجة ألعاب الفيديو لطلابها أفضل محتوى في مجال تطوير ألعاب الفيديو، وذلك بفضل تصميمها الدقيق، حيث تم تنظيمها في 10 وحدات دراسية كل منها 10 مواد. من خلالها، سيتمكن الطلاب من تعلم كل ما يحتاجونه للمشاركة في أي نوع من مشاريع ألعاب الفيديو، بحيث تكون العملية التعليمية كاملة وشاملة ومركزة تماماً على الممارسة.

المحتويات التي تحتاجها للتخصص
في برمجة ألعاب الفيديو"

الوحدة 1. أساسيات البرمجة

1.1. مقدمة في البرمجة
1.1.1. الهيكل الأساسي للكمبيوتر
2.1.1. Software
3.1.1. لغات البرمجة
4.1.1. دورة حياة تطبيقات الحاسوب
2.1. تصميم الخوارزمية
1.2.1. حل المشاكل
2.2.1. التقنيات الوصفية
3.2.1. عناصر وهيكل الخوارزمية
3.1. عناصر البرنامج
1.3.1. أصل وخصائص لغة C++
2.3.1. بيئة التطوير
3.3.1. مفهوم البرنامج
4.3.1. أنواع البيانات الأساسية
5.3.1. العاملين
6.3.1. التعبيرات
7.3.1. الأحكام
8.3.1. دخول وإخروج البيانات
4.1. أحكام التحكم
1.4.1. الأحكام
2.4.1. التشعبات
3.4.1. الحلقات
5.1. التجريد والنمطية: الوظائف
1.5.1. التصميم النموذجي
2.5.1. مفهوم الوظيفة والمنفعة
3.5.1. تعريف الخاصية
4.5.1. تدفق التنفيذ في استدعاء الخاصية
5.5.1. النموذج الأولي للخاصية
6.5.1. استرداد النتائج
7.5.1. استدعاء الخاصية: المقاييس
8.5.1. تمرير المقاييس حسب المرجع والقيمة
9.5.1. نطاق المعّرف
6.1. هياكل البيانات الثابتة
1.6.1. Arrays
2.6.1. المصفوفات. متعددات الوجوه
3.6.1. البحث والفرز
4.6.1. السلاسل. وظائف الإدخال/الإخراج للسلاسل
5.6.1. الهياكل تقاطعات
6.6.1. أنواع البيانات الجديدة
7.1. هياكل البيانات الديناميكية: المؤشرات
1.7.1. المفهوم تعريف المؤشر
2.7.1. المشغلين والعمليات مع المؤشرات
3.7.1. Arrays المؤشر
4.7.1. المؤشرات وArrays
5.7.1. مؤشرات إلى السلاسل
6.7.1. مؤشرات إلى الهياكل
7.7.1. متعددة المراوغة
8.7.1. مؤشرات الوظيفة
9.7.1. تمرير الوظائف والهياكل وArrays كمعلمات دالة
8.1. ملفات
1.8.1. مفاهيم أساسية
2.8.1. عمليات بالملفات
3.8.1. أنواع الملفات
4.8.1. تنظيم الملفات
5.8.1. مقدمة إلى ملفات C++
6.8.1. إدارة الملفات
9.1. العودية
1.9.1. تعريف التكرارية
2.9.1. أنواع التكرارية
3.9.1. المميزات والعيوب
4.9.1. الاعتبارات
5.9.1. التحويل العودي التكراري
6.9.1. مكدس العودية

10.1. الاختبار والتوثيق
1.10.1. اختبار البرنامج
2.10.1. اختبار الصندوق الأبيض
3.10.1. اختبار الصندوق الأسود
4.10.1. أدوات للاختبار
5.10.1. توثيق البرنامج

الوحدة 2. بنية البيانات والخوارزميات

1.2. مقدمة لاستراتيجيات تصميم الخوارزميات
1.1.2. العودية
2.1.2. فرق تسد
3.1.2. استراتيجيات أخرى
2.2. كفاءة وتحليل الخوارزميات
1.2.2. تدابير الكفاءة
2.2.2. قياس حجم المدخلات
3.2.2. قياس وقت التشغيل
4.2.2. أسوأ وأفضل حالة ومابينهما
5.2.2. التدوين المقارب
6.2.2. معايير التحليل الرياضي لخوارزميات السلوك الغيرالمتكرر
7.2.2. التحليل الرياضي للخوارزميات المتكررة
8.2.2. التحليل التجريبي للخوارزميات
3.2. فرز الخوارزميات
1.3.2. مفهوم الإدارة
2.3.2. فرز الفقاعة
3.3.2. الفرز حسب الاختيار
4.3.2. ترتيب الإدراج
5.3.2. الفرز حسب الخليط (Merge_Sort)
6.3.2. الفرز السريع (Quick_Sort)
4.2. خوارزميات بالأشجار
1.4.2. مفهوم الشجرة
2.4.2. أشجار ثنائية
3.4.2. جولات الأشجار
4.4.2. تمثيل التعبيرات
5.4.2. أشجار ثنائية مرتبة
6.4.2. أشجار ثنائية متوازنة

5.2. خوارزميات مع Heaps
1.5.2. الأكوام heaps
2.5.2. خوارزمية Heapsort
3.5.2. قوائم الانتظار ذات الأولوية
6.2. الخوارزميات ذات الرسوم البيانية
1.6.2. العرض
2.6.2. جولة ضيقة
3.6.2. جولة متعمقة
4.6.2. الترتيب الطوبولوجي
7.2. خوارزميات Greedy
1.7.2. استراتيجية Greedy
2.7.2. عناصر استراتيجية Greedy
3.7.2. صرف العملات
4.7.2. مشكلة المسافر
5.7.2. مشكلة حقيبة الظهر
8.2. ابحث عن الحد الأدنى من المسارات
1.8.2. مشكلة المسار الأدنى
2.8.2. الأقواس والدورات السلبية
3.8.2. خوارزمية Dijkstra
9.2. خوارزميات Greedyعلى الرسوم البيانية
1.9.2. شجرة الحد الأدنى من الطبقة
2.9.2. خوارزمية Prim
3.9.2. خوارزمية Kruskal
4.9.2. تحليل التعقيد
10.2. Backtracking
1.10.2. Backtracking
2.10.2. التقنيات البديلة

الوحدة 3. البرمجة الموجهة للكائنات

1.3. مقدمة في البرمجة الموجهة للكائنات
1.1.3. مقدمة في البرمجة الموجهة للكائنات
2.1.3. تصميم الفصول
3.1.3. مقدمة في UML لنمذجة المشاكل
2.3. العلاقات بين الفصول
1.2.3. التجريد والوراثة
2.2.3. مفاهيم الوراثة المتقدمة
3.2.3. تعدد الأشكال
4.2.3. التركيب والتجميع
3.3. مقدمة في أنماط التصميم للمشاكل الموجهة للكائنات
1.3.3. ما هي أنماط التصميم؟
2.3.3. نمط Factory
3.3.3. نمط Singleton
4.3.3. نمط Observer
5.3.3. نمط Composite
4.3. الاستثناءات
1.4.3. ما هي الاستثناءات؟
2.4.3. التقاط الاستثناءات والتعامل معها
3.4.3. رمي الاستثناءات
4.4.3. إنشاء الاستثناءات
5.3. واجهات المستخدم
1.5.3. مقدمة في Qt
2.5.3. تحديد المواقع
3.5.3. التموضع
4.5.3. ما هي الأحداث؟
5.5.3. تطوير واجهات المستخدم

الوحدة 4. وحدات تجكم وأجهزة لألعاب الفيديو

1.4. تاريخ برمجة ألعاب الفيديو
1.1.4. فترة الأتارى(1977- 1985)
2.1.4. فترة كمبيوتر العيلة ومنصة ألعاب الفيديو المنزلية NES , SNES (1985-1995)
3.1.4. فترة البلاى ستيشن 1و2 (1995-2005)
4.1.4. فترة الإكس بوكس 360 Xbox , البلاى ستيشن 3 3PS والوىWii (2005-2013)
5.1.4. فترة 4Xbox One, PS و Wii U–Switchg (2013-الحالى)
6.1.4. المستقبل
2.4. تاريخ لعب ألعاب الفيديو
1.2.4. المقدمة
2.2.4. السياق الاجتماعي
3.2.4. الرسم الهيكلى
4.2.4. مستقبل
3.4. التكيف مع العصور الحديثة
1.3.4. ألعاب قائمة على الحركة
2.3.4. الواقع الافتراضي
3.3.4. الواقع المعزز
4.3.4. الواقع المختلط
4.4. Unity: Scripting 1 والأمثلة
1.4.4. ما هى لغة البرمجة؟
2.4.4. أول لغة برمجة
3.4.4. إضافة لغة برمجة
4.4.4. فتح لغة برمجة
5.4.4. MonoBehaviour
6.4.4. Debugging
5.4. Unity: Scripting 2 أمثلة
1.5.4. إدخال لوحة المفاتيح والفارة
2.5.4. Raycast
3.5.4. عملية قراءة وتخصيص المعلومات
4.5.4. المتغيرات
5.5.4. المتغيرات العامة والمتسلسلة

6.3. مقدمة في البرمجة المتزامنة
1.6.3. مقدمة في البرمجة المتزامنة
2.6.3. مفهوم العملية والخيط
3.6.3. التفاعل بين العمليات أو الخيوط
4.6.3. الخيوط في C++
5.6.3. مزايا ومساوئ البرمجة المتزامنة
7.3. إدارة الخيط والمزامنة
1.7.3. دورة حياة الخيط
2.7.3. صنف Thread
3.7.3. جدولة الخيط
4.7.3. مجموعات الخيوط
5.7.3. المواضيع الخفية
6.7.3. التزامن
7.7.3. آليات القفل
8.7.3. آليات الاتصال
9.7.3. الشاشات
8.3. المشاكل الشائعة في البرمجة المتزامنة
1.8.3. مشكلة المنتج والمستهلك
2.8.3. مشكلة القراء والكتاب
3.8.3. مشكلة عشاء الفلاسفة
9.3. توثيق واختبار البرمجيات
1.9.3. لماذا من المهم توثيق البرمجيات؟
2.9.3. توثيق التصميم
3.9.3. استخدام أدوات التوثيق
10.3. اختبار البرمجيات
1.10.3. مقدمة في اختبار البرمجيات
2.10.3. أنواع الاختبارات
3.10.3. اختبار الوحدة
4.10.3. اختبار التكامل
5.10.3. اختبار التحقق من الصحة
6.10.3. اختبار النظام

6.4. أمثلة Unity: Scripting 3
1.6.4. الحصول على المكونات
2.6.4. تعديل المكونات
3.6.4. إختبارات
4.6.4. أشياء متعددة
5.6.4. Colliders triggers
6.6.4. الرباعيات
7.4. الملحقات
1.7.4. التطور والتصنيفات
2.7.4. الملحقات والواجهات
3.7.4. الملحقات الحالية
4.7.4. مستقبل قريب
8.4. ألعاب فيديو: وجهات نظر مستقبلية
1.8.4. الألعاب القائمة على التحديث
2.8.4. غياب أدوات التحكم
3.8.4. واقع غامر
4.8.4. بدائل أخرى
9.4. البنيات
1.9.4. إحتياجات خاصة فى العاب الفيديو
2.9.4. تطور الهندسة المعمارية لألعاب الفيديو
3.9.4. هندسة معمارية حالية
4.9.4. إختلافات بين الهندسة المعمارية لألعاب الفيديو
10.4. تطور أدوات التطور
1.10.4. المقدمة
2.10.4. الجيل الثالث لادوات التطور
3.10.4. الجيل الرابع لأدوات التطور
4.10.4. الجيل الخامس لأدوات التطور
5.10.4. الجيل السادس لأدوات التطور

الوحدة 5. هندسة البرمجيات

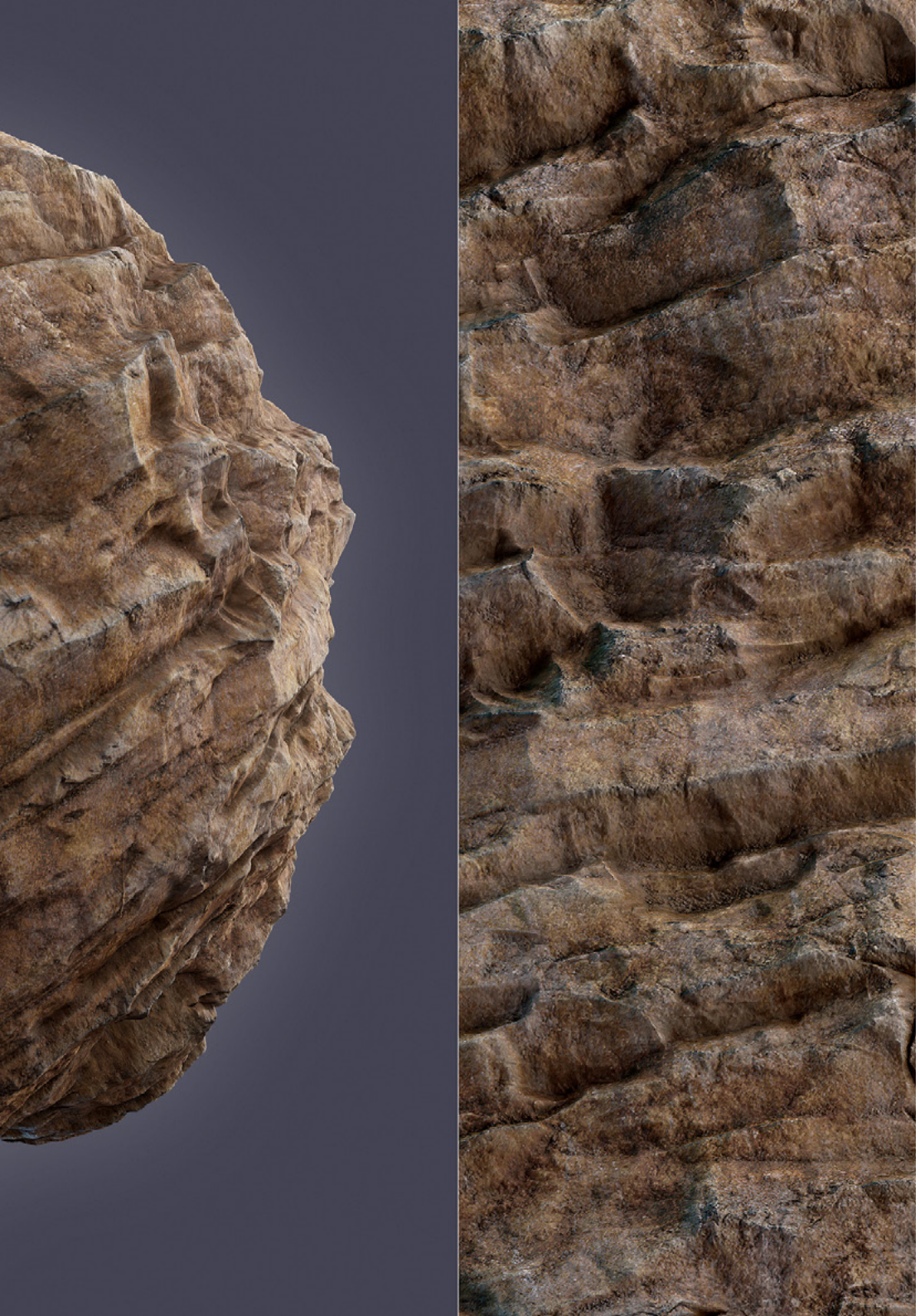
1.5. مقدمة في هندسة البرمجيات والنمذجة
1.1.5. طبيعة البرمجيات
2.1.5. الطبيعة الفريدة لتطبيقات الويب
3.1.5. هندسة البرمجيات
4.1.5. عملية البرمجيات
5.1.5. ممارسة هندسة البرمجيات
6.1.5. أساطير البرمجيات
7.1.5. كيف يبدأ كل شيء؟
8.1.5. المفاهيم الموجهة للكائنات
9.1.5. مقدمة في UML
2.5. عملية البرمجيات
1.2.5. نموذج عملية عامة
2.2.5. نماذج عملية إرشادية
3.2.5. نماذج العمليات المتخصصة
4.2.5. العملية الموحدة
5.2.5. نماذج العمليات الشخصية والجماعية
6.2.5. ما هي خفة الحركة؟
7.2.5. ما هي العملية الرشيقة؟
8.2.5. Scrum
9.2.5. مجموعة أدوات العملية الرشيقة
3.5. المبادئ التوجيهية لممارسة هندسة البرمجيات
1.3.5. المبادئ الموجهة للعملية
2.3.5. المبادئ الموجهة للممارسة
3.3.5. مبادئ التواصل
4.3.5. مبادئ التخطيط
5.3.5. مبادئ النمذجة
6.3.5. مبادئ البناء
7.3.5. مبادئ النشر

4.5. فهم المتطلبات
1.4.5. هندسة المتطلبات
2.4.5. إنشاء الأساس
3.4.5. استنباط المتطلبات
4.4.5. تطوير حالات الاستخدام
5.4.5. نمذجة المتطلبات
6.4.5. التفاوض على المتطلبات
7.4.5. التحقق من المتطلبات
5.5. نمذجة المتطلبات: السيناريوهات والمعلومات وفئات التحليل
1.5.5. تحليل المتطلبات
2.5.5. النمذجة القائمة على السيناريو
3.5.5. نماذج UML التي توفر حالة الاستخدام
4.5.5. مفاهيم نمذجة البيانات
5.5.5. النمذجة القائمة على الفئات
6.5.5. مخططات الفئات
6.5. نمذجة المتطلبات: التدفق والسلوك والأنماط
1.6.5. استراتيجيات نمذجة المتطلبات
2.6.5. النمذجة الموجهة نحو التدفق
3.6.5. مخططات الحالة
4.6.5. إنشاء نموذج سلوكي
5.6.5. مخططات التسلسل البياني
6.6.5. مخططات التواصل
7.6.5. أنماط نمذجة المتطلبات
7.5. مفاهيم التصميم
1.7.5. التصميم في سياق هندسة البرمجيات
2.7.5. عملية التصميم
3.7.5. مفاهيم التصميم
4.7.5. مفاهيم التصميم الموجه للكائنات
5.7.5. نموذج التصميم

8.5. تصميم الهندسة
1.8.5. هندسة البرمجيات
2.8.5. الأنواع الهندسية
3.8.5. الأنماط الهندسية
4.8.5. التصميم الهندسي
5.8.5. تطور التصاميم البديلة للهندسة
6.8.5. تخطيط البنية باستخدام تدفقات البيانات
9.5. التصميم على مستوى المكونات والتصميم القائم على الأنماط
1.9.5. ما هو المكون؟
2.9.5. تصميم المكوّنات على أساس الصنف
3.9.5. تحقيق التصميم على مستوى المكوّن
4.9.5. تصميم المكوّنات التقليدي
5.9.5. التطوير القائم على المكونات
6.9.5. أنماط التصميم
7.9.5. تصميم البرمجيات القائم على الأنماط
8.9.5. الأنماط الهندسية
9.9.5. أنماط تصميم مستوى المكونات
10.9.5. أنماط تصميم واجهة المستخدم
10.5. جودة البرمجيات وإدارة المشاريع
1.10.5. الجودة
2.10.5. جودة البرمجيات
3.10.5. معضلة جودة البرمجيات
4.10.5. تحقيق جودة البرمجيات
5.10.5. ضمان جودة البرمجيات
6.10.5. الطيف الإداري
7.10.5. الموظفين
8.10.5. المنتج
9.10.5. العملية
10.10.5. المشروع
11.10.5. المبادئ والممارسات

الوحدة 6. الموتورات المحركة للالعاب الفيديو

1.6. ألعاب الفيديو وتكنولوجيا الإتصالات والمعلومات TICs
1.1.6. المقدمة
2.1.6. الفرص
3.1.6. التحديات
4.1.6. الاستنتاجات
2.6. تاريخ محركات ألعاب الفيديو
1.2.6. المقدمة
2.2.6. فترة الأتارى
3.2.6. فترة الثمانيات
4.2.6. المحركات الأولى فترة التسعينات
5.2.6. المحركات الحالية
3.6. الموتورات المحركة للالعاب الفيديو
1.3.6. أنواع المحركات
2.3.6. أجزاء من الموتور المحرك ألعاب الفيديو
3.3.6. المحركات الحالية
4.3.6. اختيار محرك لمشروعنا
4.6. محرك Game Maker
1.4.6. المقدمة
2.4.6. تصميم خشبة المسرح الإفتراضية
3.4.6. الرسوم المتحركةSprites
4.4.6. التصادمات
5.4.6. البرمجة النصية فى لغة العلامات الجغرافية GML
5.6. المحرك الغير واقعى 4: مقدمة
1.5.6. ما هو المحرك الغير واقعى 4؟ ما هى فلسفته؟
2.5.6. المعدات
3.5.6. UI
4.5.6. التنشيط
5.5.6. نظام الجسيمات
6.5.6. الذكاء الاصطناعي
7.5.6. معدل الإطارات التى تعرض فى الثانية الواحدة FPS



6.6. المحرك الغير واقعى 4: Visual Scripting
1.6.6. فلسفة المخططات والبرمجة البصرية
2.6.6. Debugging
3.6.6. أنواع المتغيرات
4.6.6. التحكم في تدفق المواد
7.6. Motor Unity 5
1.7.6. برمجة فى C # والأستوديو البصرى
2.7.6. إنشاء المنازل مسبقة الصنع
3.7.6. استخدام Gizmos للتحكم فى ألعاب الفيديو
4.7.6. المحرك المتكيف: ثنائية الأبعاد وثلاثية الابعاد
8.6. المحرك Godot
1.8.6. فلسفة تصميم Godot
2.8.6. تصميم موجه للأدوات والتركيب
3.8.6. كل شيء مدرج في حزمة واحدة
4.8.6. سوفت وير حر وبقوده المجتمع
9.6. الموتور المحرك صانع الألعاب ثنائية الأبعاد من طراز RPG
1.9.6. فلسفة صانع فيديو تقمص الأدوار RPG Maker
2.9.6. الاخذ كمرجع
3.9.6. إنشاء لعبة ذات شخصية
4.9.6. ألعاب تجارية ناجحة
10.6. المحرك مصدر 2
1.10.6. فلسفة المصدر 2
2.10.6. المصدر ومصدر 2: التقييم
3.10.6. استخدام مجتمع ألعاب الفيديو: محتوى سمعى بصرى وألعاب فيديو
4.10.6. مستقبل المحرك مصدر 2
5.10.6. التعديلات والألعاب الناجحة

الوحدة 7. أنظمة ذكية

1.7. نظرية الوكلاء
1.1.7. تاريخ المفهوم
2.1.7. تعريف الوكلاء
3.1.7. وكلاء في الذكاء الاصطناعي
4.1.7. وكلاء في هندسة البرمجيات
2.7. بناء الوكلاء
1.2.7. عملية التفكير في عامل ما
2.2.7. عوامل تفاعلية
3.2.7. العوامل الاستنتاجية
4.2.7. عوامل هجينة
5.2.7. مقارنة
3.7. المعلومات والمعارف
1.3.7. التمييز بين البيانات والمعلومات والمعارف
2.3.7. تقييم جودة البيانات
3.3.7. طرائق جمع البيانات
4.3.7. طرائق الحصول على المعلومات
5.3.7. طرائق اكتساب المعرفة
4.7. تمثيل المعارف
1.4.7. أهمية تمثيل المعارف
2.4.7. تعريف تمثيل المعرفة من خلال أدوارها
3.4.7. خصائص تمثيل المعرفة
5.7. علم المعلومات
1.5.7. مقدمة للبيانات الوصفية
2.5.7. المفهوم الفلسفي لعلم الأنطولوجيا
3.5.7. مفهوم الحاسوب لعلم الأنطولوجيا
4.5.7. أنطولوجيات المجال وأنطولوجيات المستوى الأعلى
5.5.7. كيف تبني أنطولوجيا؟

6.7. اللغات الوجودية والبرمجيات لإنشاء الأنطولوجيا
1.6.7. قوائم RDF وTurtle و 3N
2.6.7. RDF Schema
3.6.7. OWL
4.6.7. SPARQL
5.6.7. مقدمة إلى الأدوات المختلفة لإنشاء الأنطولوجيا
6.6.7. تركيب واستخدام Protégé
7.7. الويب الدلالي
1.7.7. الحالة الحالية والمستقبلية للشبكة الدلالية
2.7.7. تطبيقات الشبكة الدلالية
8.7. نماذج أخرى لتمثيل المعرفة
1.8.7. المفردات
2.8.7. نظرة عامة
3.8.7. التصنيفات
4.8.7. المرادفات
5.8.7. فولكسونومي
6.8.7. مقارنة
7.8.7. خرائط العقل
9.7. تقييم وإدماج التمثيلات المعرفية
1.9.7. منطق الترتيب الصفري
2.9.7. المنطق من الدرجة الأولى
3.9.7. المنطق الوصفي
4.9.7. العلاقة بين مختلف أنواع المنطق
5.9.7. Prolog: برمجة تعتمد على منطق الدرجة الأولى
10.7. المسبوبون الدلاليون والأنظمة المبنية على المعرفة والأنظمة الخبيرة
1.10.7. مفهوم المنطق
2.10.7. طلبات المعقل
3.10.7. النظم القائمة على المعرفة
4.10.7. MYCIN، تاريخ أنظمة الخبراء
5.10.7. عناصر وبناء نظام الخبراء
6.10.7. إنشاء الأنظمة المتخصصة

الوحدة 8. البرمجة في الوقت الفعلى

1.8. مفاهيم أساسيات البرمجة المتزامنة
1.1.8. مفاهيم اساسية
2.1.8. التزامن
3.1.8. فوائد التزامن
4.1.8. التزامن، hardware
2.8. الهياكل الأساسية لدعم التزامن الأساسى في Java
1.2.8. التزامن فى Java
2.2.8. إنشاء Threads
3.2.8. مناهج
4.2.8. التزامن
3.8. Threads, دورة الحياة، الأولويات، المقاطعات، الحالات، المنفذين
1.3.8. Threads
2.3.8. دورة الحياة
3.3.8. الأولويات
4.3.8. المقاطعات
5.3.8. ولايات
6.3.8. المنفذين
4.8. الاستبعاد المتبادل
1.4.8. ما هو الاستبعاد المتبادل؟
2.4.8. خوارزمية Dekker
3.4.8. خوارزمية Peterson
4.4.8. الاستبعاد المتبادل فى Java
5.8. تبعيات الولايات
1.5.8. حقن التبعيات
2.5.8. تنفيذ النمط فى Java
3.5.8. أشكال حقن التبعيات
4.5.8. مثال

الوحدة 9. تصميم وتطوير ألعاب الويب

1.9. أصول ومعايير الويب
1.1.9. أصول الإنترنت
2.1.9. إنشاء World Wide Web
3.1.9. ظهور معايير الويب
4.1.9. ظهور معايير الويب
2.9. بروتوكول نقل النص الفائق HTTP وهيكل خادم العميل
1.2.9. دور خادم ـ عميل
2.2.9. اتصالات خادم ـ عميل
3.2.9. التاريخ الحديث
4.2.9. الحوسبة المركزية
3.9. برمجة الويب: مقدمة
1.3.9. مفاهيم أساسية
2.3.9. إعداد خادم الويب
3.3.9. المفاهيم الأساسية فى 5HTML
4.3.9. أشكال HTML
4.9. مقدمة في HTML و أمثلة
1.4.9. تاريخ 5HTML
2.4.9. عناصر 5HTML
3.4.9. APIs
4.4.9. 3CCS:
5.9. نموذج كائن المستند
1.5.9. ما هو نموذج كائن المستند؟
2.5.9. استخدام DOCTYPE
3.5.9. أهمية التحقق من صحة HTML
4.5.9. الوصول إلى العناصر
5.5.9. إنشاء العناصر والنصوص
6.5.9. استخدام innerHTML
7.5.9. حذف عنصر أو عقدة نصية
8.5.9. قراءة وكتابة خصائص العنصر
9.5.9. التعامل مع أنماط العناصر
10.5.9. إرفاق ملفات متعددة في وقت واحد

6.8. أنماط التصميم
1.6.8. المقدمة
2.6.8. أنماط الإنشاء
3.6.8. أنماط البناء
4.6.8. أنماط التصرف
7.8. استخدام المكتبة Java
1.7.8. ما هى المكتبات فى Java ؟
2.7.8. Mockito-all, Mockito-core
3.7.8. Guava
4.7.8. Commons-io
5.7.8. 3Commons-lang, Commons-lang
8.8. برمجة Shaders
1.8.8. Pipeline التنقيط وثلاثى الأبعاد
2.8.8. Vertex Shading
3.8.8. Pixel Shading: الإضاءة 1
4.8.8. Pixel Shading: الإضاءة 2
5.8.8. التأثيرات المترتبة
9.8. البرمجة في الوقت الفعلى
1.9.8. المقدمة
2.9.8. مقاطعات المعالجة
3.9.8. التزامن والتواصل بين العمليات
4.9.8. أنظمة التخطيط في في الوقت الفعلي
10.8. التخطيط فى الوقت الفعلى
1.10.8. المفاهيم
2.10.8. النموذج المرجعي لأنظمة الوقت الفعلى
3.10.8. سياسات التخطيط
4.10.8. المخططون الدوريون
5.10.8. المخططون بخصائص ثابتة
6.10.8. المخططون بخصائص الديناميكا

6.9. مقدمة في CSS و أمثلة
1.6.9. بناء الجملة 3CSS
2.6.9. أوراق النمط
3.6.9. الملصقات
4.6.9. محددات
5.6.9. تصميم الويب باستخدام CSS
7.9. مقدمة وأمثلة فى JavaScript
1.7.9. ما هو JavaScript ؟
2.7.9. تاريخ موجز للغة
3.7.9. إصدارات JavaScript
4.7.9. إظهار مربع حوار
5.7.9. بناء الجملة فى JavaScript
6.7.9. فهم Scripts
7.7.9. المساحات
8.7.9. تعليقات
9.7.9. المهام
10.7.9. JavaScript فى الصفحة وخارجى
8.9. مهام JavaScript
1.8.9. تصريحات المهنة
2.8.9. تعابير المهنة
3.8.9. الدعوة إلى المهام
4.8.9. العودية
5.8.9. وظائف متداخلة وإغلاقات
6.8.9. الحفاظ المتغير
7.8.9. وظائف متعددة العناوين
8.8.9. تعارضات الأسماء
9.8.9. عمليات الإغلاق والإغلاقات
10.8.9. مقاييس الوظيفة

9.9. PlayCanvas لتطوير ألعاب الويب
1.9.9. ما هو PlayCanvas ؟
2.9.9. إعدادات المشروع
3.9.9. إنشاء أداة
4.9.9. إضافات الأشياء المادية
5.9.9. إضافة النموذج
6.9.9. تغيير إعدادات الجاذبية والمشهد
7.9.9. تنفيذ Scripts
8.9.9. التحكم بالكاميرا
10.9. Phaser لتطوير ألعاب الويب
1.10.9. ما هو Phaser ؟
2.10.9. تحميل موارد
3.10.9. بناء العالم
4.10.9. المنصات الإفتراضية
5.10.9. اللاعب
6.10.9. إضافة الأشياء المادية
7.10.9. استخدام لوحة المفاتيح
8.10.9. التقاط pickups
9.10.9. النقاط وعلامات الترقيم
10.10.9. قنابل ارتدادية

الوحدة 10. الشبكات والأنظمة متعددة اللاعبين

1.10. تاريخ وتطور ألعاب الفيدو متعددة اللاعبين
1.1.10. السبعينيات 1970: أول ألعاب متعددة اللاعبين
2.1.10. سنة 90: كوايك Quake ، دوم Doom ، دوك نوكيم Duke Nukem
3.1.10. ظهور ألعاب الفيديو متعددة اللاعبين
4.1.10. متعدد اللاعبين محليًا وعبر الإنترنت
5.1.10. ألعاب الحفلة

2.10. نماذج أعمال الفيديو التجارية متعددة اللاعبين
1.2.10. أصل وتشغيل نماذج أعمال الفيديو التجارية الناشئة
2.2.10. خدمات للبيع عبر الإنترنت
3.2.10. لعب مجانى
4.2.10. العاب بالدفع القليل
5.2.10. الدعاية
6.2.10. الإشتراك مع الدفع الشهرى
7.2.10. الدفع لكل لعبة
8.2.10. تجربة مجانية قبل الشراء
3.10. ألعاب محلية وألعاب عير شبكات التواصل
1.3.10. ألعاب محلية: بدايات
2.3.10. ألعاب الحفلة نينتندو ووحدة الأسرة
3.3.10. ألعاب شبكات التواصل: بدايات
4.3.10. تقييم ألعاب شبكات الواصل
4.10. نموذج النظام المترابط المفتوح OSI: الطبقات
1.4.10. نموذج الربط البينى للأنظمة المفتوحة OSI: مقدمة
2.4.10. طبقة مادية
3.4.10. طبقة رابط البيانات
4.4.10. طبقة الشبكة
5.10. نموذج النظام المترابط المفتوح OSI: الطبقات II
1.5.10. طبقة النقل
2.5.10. طبقة الجلسة
3.5.10. طبقة المقدمة
4.5.10. طبقة التطبيق

6.10. شبكات الحواسيب والإنترنت
1.6.10. ما هي شبكة الحواسيب؟
2.6.10. Software
3.6.10. المكونات المادية للحاسوب Hardware
4.6.10. الخوادم
5.6.10. تخزين الشبكة
6.6.10. بروتوكوليات الشبكة
7.10. شبكات الهواتف النقالة واللاسلكية
1.7.10. شبكة الهواتف النقالة
2.7.10. شبكة لاسلكية
3.7.10. تشغيل شبكة الهواتف النقالة
4.7.10. التقنية الرقمية
8.10. الأمان
1.8.10. أمان شخصى
2.8.10. الإختراقات و الغش فى ألعاب الفيديو
3.8.10. أمان ضد الأفخاخ
4.8.10. تحليل أنظمة أمن مكافحة الغش
9.10. أنظمة متعددة اللاعبين: خوادم
1.9.10. إقامة الخوادم
2.9.10. ألعاب الفيديو الضخمة متعددة اللاعبين MMO
3.9.10. خوادم مخصصة لألعاب الفيديو
4.9.10. الشبكات المحلية LAN الحفلات
10.10. تصميم وبرمجة ألعاب الفيديو متعددة اللاعبين
1.10.10. أساس تصميم ألعاب الفيديو متعددة اللاعبين فى اللاواقعية
2.10.10. أساس تصميم ألعاب الفيديو متعددة اللاعبين فى الوحدة
3.10.10. كيفية جعل لعبة متعددة اللاعبين ممتعة
4.10.10. ما وراء وحدة التحكم: الابتكار في عناصر التحكم متعددة اللاعبين

05

المنهجية

يقدم هذا البرنامج التدريبي طريقة مختلفة للتعلم. فقد تم تطوير منهجيتنا من خلال أسلوب التعليم المرتكز على التكرار: ***Relearning*** أو ما يعرف بمنهجية إعادة التعلم.

يتم استخدام نظام التدريس هذا، على سبيل المثال، في أكثر كليات الطب شهرة في العالم، وقد تم اعتباره أحد أكثر المناهج فعالية في المنشورات ذات الصلة مثل مجلة نيو إنجلند الطبية ***(New England Journal of Medicine)***.

اكتشف منهجية *Relearning* (منهجية إعادة التعلم)، وهي نظام يتخلى عن التعلم الخطي التقليدي ليأخذك عبر أنظمة التدريس التعليم المرتكزة على التكرار: إنها طريقة تعلم أثبتت فعاليتها بشكل كبير، لا سيما في المواد الدراسية التي تتطلب الحفظ"

منهج دراسة الحالة لوضع جميع محتويات المنهج في سياقها المناسب

يقدم برنامجنا منهج ثوري لتطوير المهارات والمعرفة. هدفنا هو تعزيز المهارات في سياق متغير وتنافسي ومتطلب للغاية.

> مع جامعة TECH يمكنك تجربة طريقة تعلم تهز أسس الجامعات التقليدية في جميع أنحاء العالم"

سيتم توجيهك من خلال نظام التعلم القائم على إعادة التأكيد على ما تم تعلمه، مع منهج تدريس طبيعي وتقدمي على طول المنهج الدراسي بأكمله.

سيتعلم الطالب، من خلال الأنشطة التعاونية والحالات الحقيقية، حل المواقف المعقدة في بيئات العمل الحقيقية.

منهج تعلم مبتكرة ومختلفة

إن هذا البرنامج المُقدم من خلال TECH هو برنامج تدريس مكثف، تم خلقه من الصفر، والذي يقدم التحديات والقرارات الأكثر تطلبًا في هذا المجال، سواء على المستوى المحلي أو الدولي. تعزز هذه المنهجية النمو الشخصي والمهني، متخذة بذلك خطوة حاسمة نحو تحقيق النجاح. ومنهج دراسة الحالة، وهو أسلوب يرسي الأسس لهذا المحتوى، يكفل اتباع أحدث الحقائق الاقتصادية والاجتماعية والمهنية.

> يعدك برنامجنا هذا لمواجهة تحديات جديدة في بيئات غير مستقرة ولتحقيق النجاح في حياتك المهنية "

كان منهج دراسة الحالة هو نظام التعلم الأكثر استخدامًا من قبل أفضل كليات إدارة الأعمال في العالم منذ نشأتها. تم تطويره في عام 1912 بحيث لا يتعلم طلاب القانون القوانين بناءً على المحتويات النظرية فحسب، بل اعتمد منهج دراسة الحالة على تقديم مواقف معقدة حقيقية لهم لاتخاذ قرارات مستنيرة وتقدير الأحكام حول كيفية حلها. في عام 1924 تم تحديد هذه المنهجية كمنهج قياسي للتدريس في جامعة هارفارد.

أمام حالة معينة، ما الذي يجب أن يفعله المهني؟ هذا هو السؤال الذي سنواجهك بها في منهج دراسة الحالة، وهو منهج تعلم موجه نحو الإجراءات المتخذة لحل الحالات. طوال أربع سنوات البرنامج، ستواجه عدة حالات حقيقية. يجب عليك دمج كل معارفك والتحقيق والجدال والدفاع عن أفكارك وقراراتك.

منهجية إعادة التعلم *(Relearning)*

تجمع جامعة TECH بين منهج دراسة الحالة ونظام التعلم عن بعد، 100٪ عبر الانترنت والقائم على التكرار، حيث تجمع بين 8 عناصر مختلفة في كل درس.

نحن نعزز منهج دراسة الحالة بأفضل منهجية تدريس 100٪ عبر الانترنت في الوقت الحالي وهي: منهجية إعادة التعلم والمعروفة بـ *Relearning*.

في عام 2019، حصلنا على أفضل نتائج تعليمية متفوقين بذلك على جميع الجامعات الافتراضية الناطقة باللغة الإسبانية في العالم.

في TECH ستتعلم بمنهجية رائدة مصممة لتدريب مدراء المستقبل. وهذا المنهج، في طليعة التعليم العالمي، يسمى *Relearning* أو إعادة التعلم.

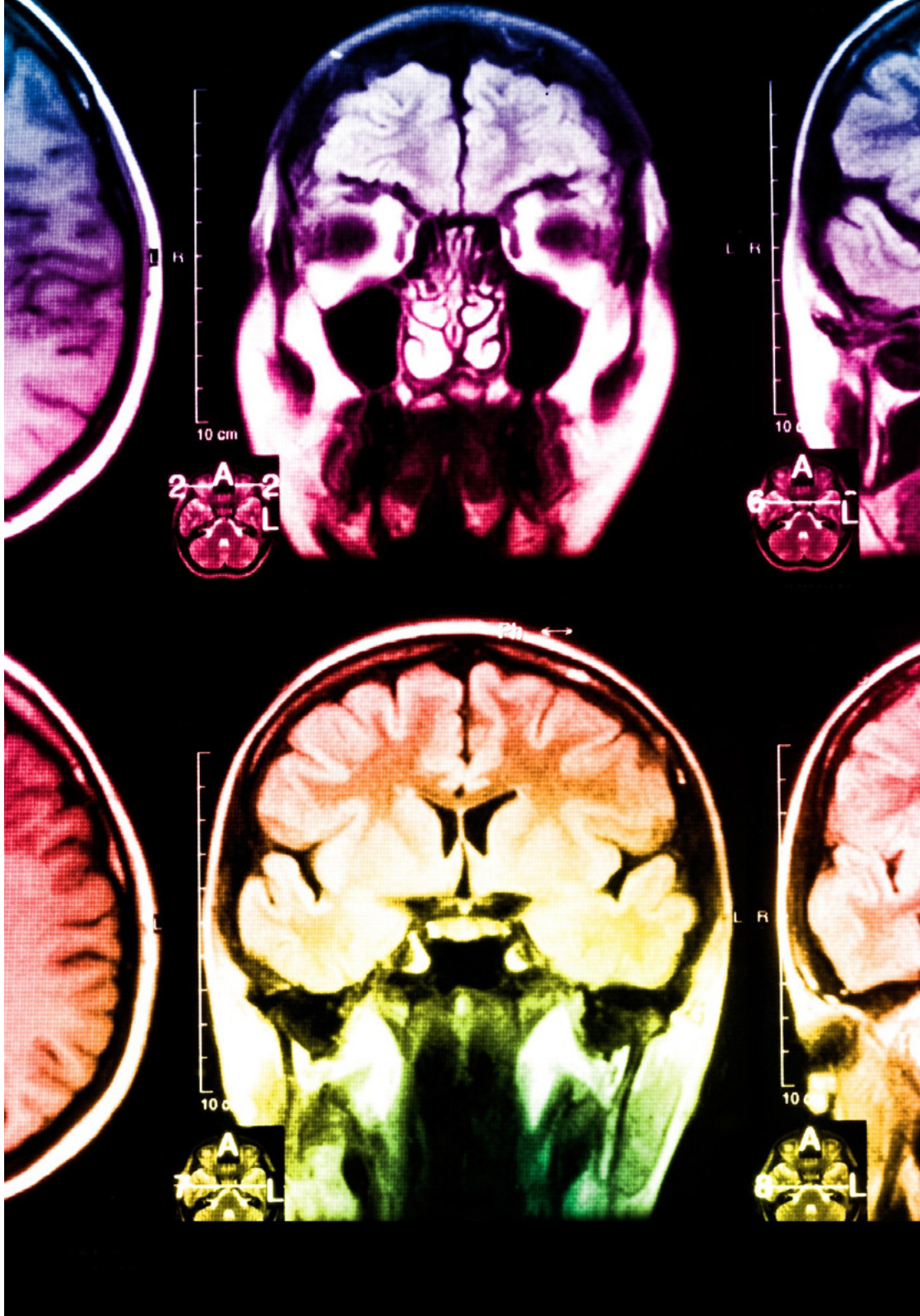
جامعتنا هي الجامعة الوحيدة الناطقة باللغة الإسبانية المصرح لها لاستخدام هذا المنهج الناجح. في عام 2019، تمكنا من تحسين مستويات الرضا العام لطلابنا من حيث (جودة التدريس، جودة المواد، هيكل الدورة، الأهداف..) فيما يتعلق بمؤشرات أفضل جامعة عبر الإنترنت باللغة الإسبانية.

في برنامجنا، التعلم ليس عملية خطية، ولكنه يحدث في شكل لولبي (نتعلّم ثم نطرح ماتعلمناه جانبًا فننساه ثم نعيد تعلمه). لذلك، نقوم بدمج كل عنصر من هذه العناصر بشكل مركزي. باستخدام هذه المنهجية، تم تدريب أكثر من 650000 خريج جامعي بنجاح غير مسبوق في مجالات متنوعة مثل الكيمياء الحيوية، وعلم الوراثة، والجراحة، والقانون الدولي، والمهارات الإدارية، وعلوم الرياضة، والفلسفة، والقانون، والهندسة، والصحافة، والتاريخ، والأسواق والأدوات المالية. كل ذلك في بيئة شديدة المتطلبات، مع طلاب جامعيين يتمتعون بمظهر اجتماعي واقتصادي مرتفع ومتوسط عمر يبلغ 43.5 عاماً.

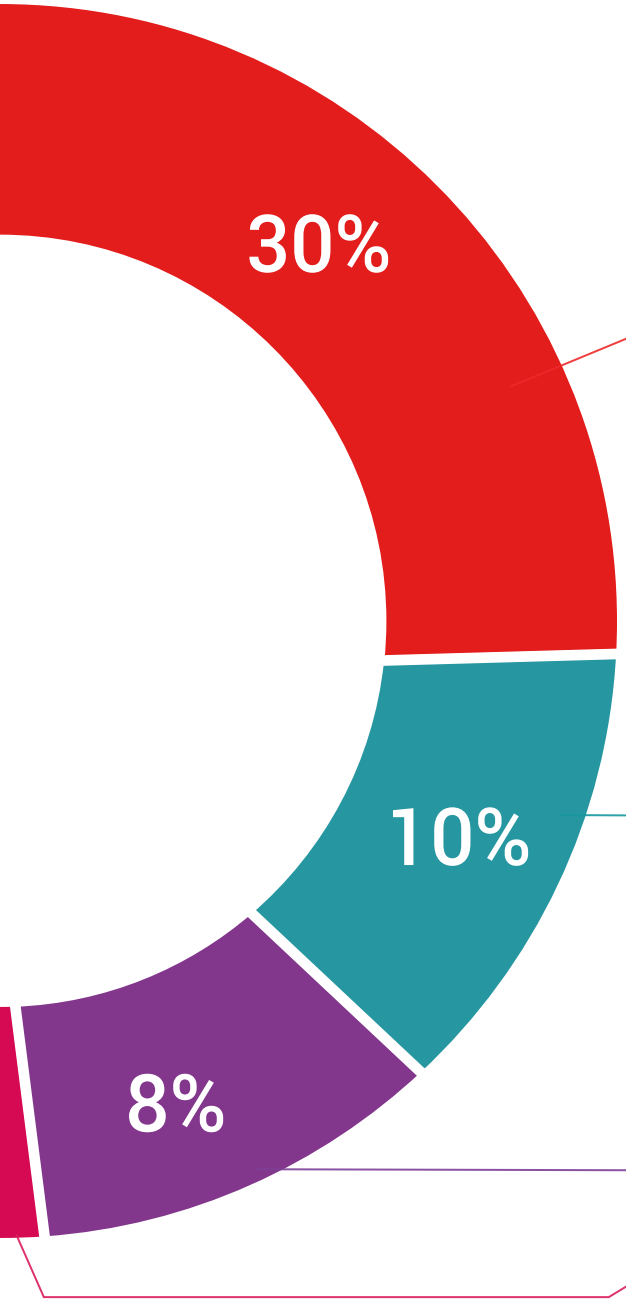
ستتيح لك منهجية إعادة التعلم والمعروفة بـ *Relearning*، التعلم بجهد أقل ومزيد من الأداء، وإشراكك بشكل أكبر في تدريبك، وتنمية الروح النقدية لديك، وكذلك قدرتك على الدفاع عن الحجج والآراء المتباينة: إنها معادلة واضحة للنجاح.

استنادًا إلى أحدث الأدلة العلمية في مجال علم الأعصاب، لا نعرف فقط كيفية تنظيم المعلومات والأفكار والصور والذكريات، ولكننا نعلم أيضًا أن المكان والسياق الذي تعلمنا فيه شيئًا هو ضروريًا لكي نكون قادرين على تذكرها وتخزينها في الحُصين بالمخ، لكي نحتفظ بها في ذاكرتنا طويلة المدى.

بهذه الطريقة، وفيما يسمى التعلم الإلكتروني المعتمد على السياق العصبي، ترتبط العناصر المختلفة لبرنامجنا بالسياق الذي يطور فيه المشارك ممارسته المهنية.



يقدم هذا البرنامج أفضل المواد التعليمية المُعَدَّة بعناية للمهنيين:



المواد الدراسية

يتم إنشاء جميع محتويات التدريس من قبل المتخصصين الذين سيقومون بتدريس البرنامج الجامعي، وتحديداً من أجله، بحيث يكون التطوير التعليمي محددًا وملموسًا حقًا.

ثم يتم تطبيق هذه المحتويات على التنسيق السمعي البصري الذي سيخلق منهج جامعة TECH في العمل عبر الإنترنت. كل هذا بأحدث التقنيات التي تقدم أجزاء عالية الجودة في كل مادة من المواد التي يتم توفيرها للطالب.

المحاضرات الرئيسية

هناك أدلة علمية على فائدة المراقبة بواسطة الخبراء كطرف ثالث في عملية التعلم.

إن مفهوم ما يسمى *Learning from an Expert* أو التعلم من خبير يقوي المعرفة والذاكرة، ويولد الثقة في القرارات الصعبة في المستقبل.

التدريب العملي على المهارات والكفاءات

سيقومون بتنفيذ أنشطة لتطوير مهارات وقدرات محددة في كل مجال مواضيعي. التدريب العملي والديناميكيات لاكتساب وتطوير المهارات والقدرات التي يحتاجها المتخصص لنموه في إطار العولمة التي نعيشها.

قراءات تكميلية

المقالات الحديثة، ووثائق اعتمدت بتوافق الآراء، والأدلة الدولية..من بين آخرين. في مكتبة جامعة TECH الافتراضية، سيتمكن الطالب من الوصول إلى كل ما يحتاجه لإكمال تدريبه.

دراسات الحالة (Case studies)

سيقومون بإكمال مجموعة مختارة من أفضل دراسات الحالة المختارة خصيصًا لهذا المؤهل.
حالات معروضة ومحللة ومدروسة من قبل أفضل المتخصصين على الساحة الدولية.

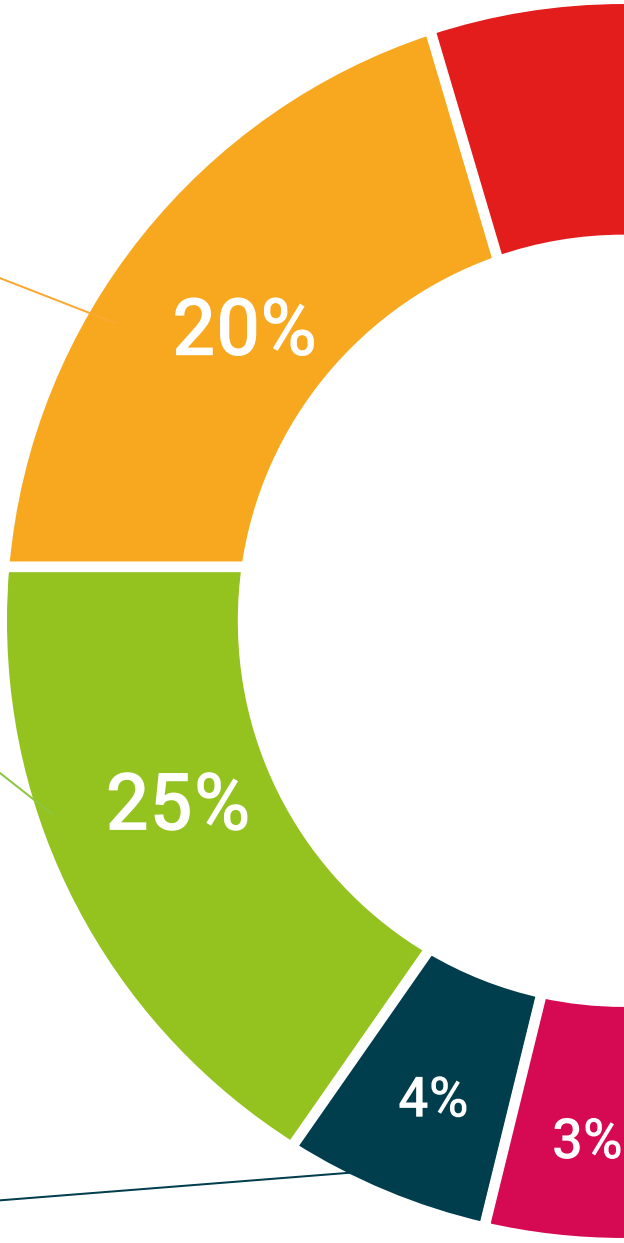
ملخصات تفاعلية

يقدم فريق جامعة TECH المحتويات بطريقة جذابة وديناميكية في أقراص الوسائط المتعددة التي تشمل الملفات الصوتية والفيديوهات والصور والرسوم البيانية والخرائط المفاهيمية من أجل تعزيز المعرفة.

اعترفت شركة مايكروسوف بهذا النظام التعليمي الفريد لتقديم محتوى الوسائط المتعددة على أنه "قصة نجاح أوروبية"

الاختبار وإعادة الاختبار

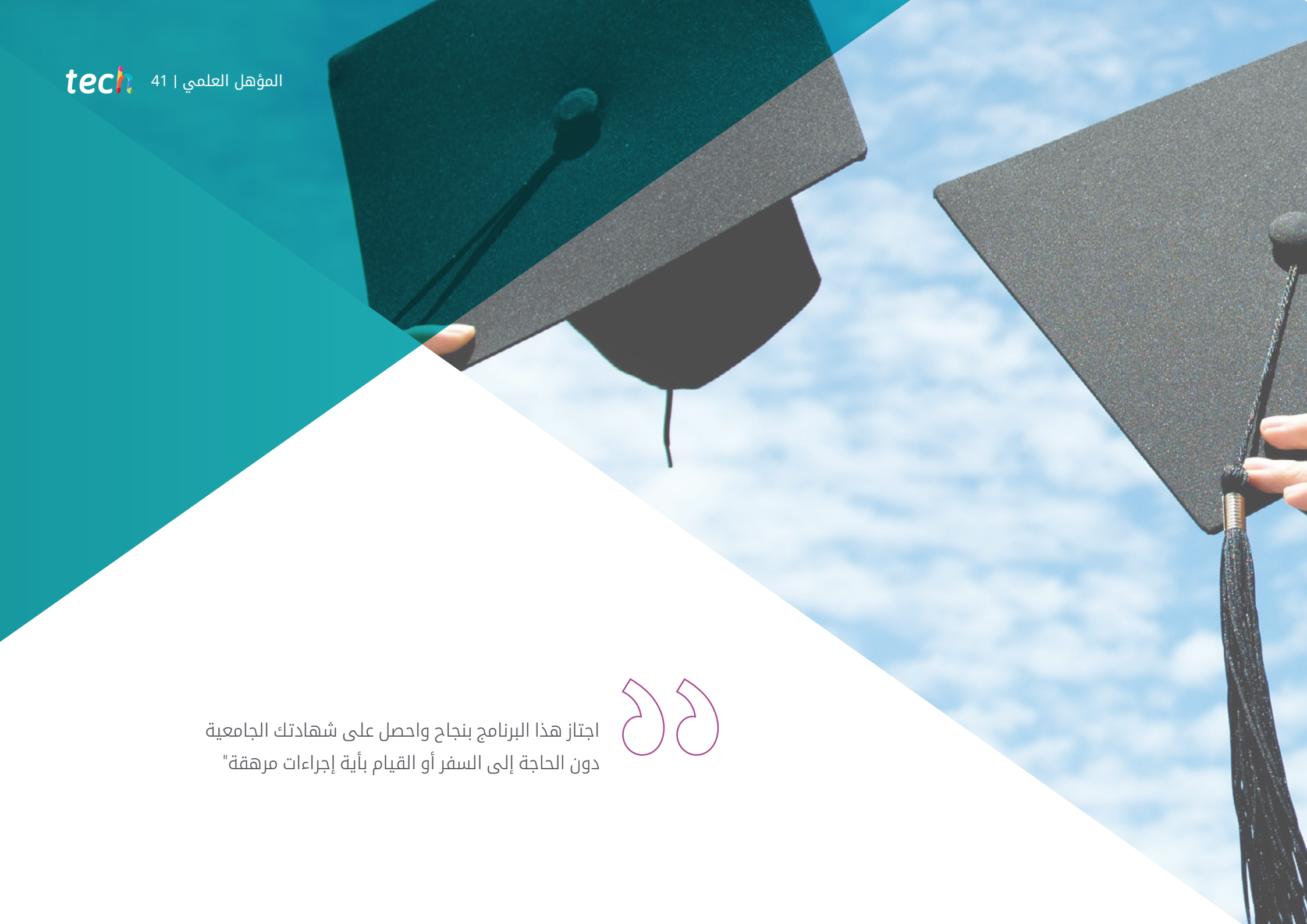
يتم بشكل دوري تقييم وإعادة تقييم معرفة الطالب في جميع مراحل البرنامج، من خلال الأنشطة والتدريبات التقييمية وذاتية التقييم: حتى يتمكن من التحقق من كيفية تحقيق أهدافه.



06

المؤهل العلمى

تضمن درجة الماجستير الخاص في برمجة ألعاب الفيديو بالإضافة إلى التدريب الأكثر دقة وحداثة، الحصول على مؤهل الماجستير الخاص الصادر عن TECH الجامعة التكنولوجية.



اجتاز هذا البرنامج بنجاح واحصل على شهادتك الجامعية
دون الحاجة إلى السفر أو القيام بأية إجراءات مرهقة"

تحتوي درجة **الماجستير الخاص في برمجة ألعاب الفيدي** على البرنامج العلمي الأكثر اكتمالا وحداثة في السوق.

بعد اجتياز التقييم، سيحصل الطالب عن طريق البريد العادي* مصحوب بعلم وصول مؤهل **الماجستير الخاص** الصادرعن **TECH الجامعة التكنولوجية**.

إن المؤهل الصادرعن **TECH الجامعة التكنولوجية** سوف يشير إلى التقدير الذي تم الحصول عليه في برنامج الماجستير الخاص وسوف يفي بالمتطلبات التي عادة ما تُطلب من قبل مكاتب التوظيف ومسابقات التعيين ولجان التقييم الوظيفي والمهني.

المؤهل العلمي: **ماجيستير خاص في برمجة ألعاب الفيدي**

طريقة الدراسة: **عبر الإنترنت**

مدة الدراسة: **12 شهر**

tech الجامعة التكنولوجية

شهادة تخرج

هذه الشهادة ممنوحة إلى

لـ

المواطن/المواطنة ..... مع وثيقة تحقيق شخصية رقم .....

لاجتيازه/لاجتيازها بنجاح والحصول على برنامج

ماجستير خاص

في

ماجستير خاص في

وهي شهادة خاصة من هذه الجامعة موافقة لـ 1500
ساعة، مع تاريخ بدء يوم/شهر/ سنة وتاريخ انتهاء يوم/شهر/سنة

تيك مؤسسة خاصة للتعليم العالي معتمدة من وزارة التعليم العام منذ 28 يونيو 2018

في تاريخ 17 يونيو 2020

أ. د./ Tere Guevara Navarro
رئيس الجامعة

الكود الفريد الخاص بجامعة TECH: AFWOR23S techtitute.com/certificates

ماجستير خاص في برمجة ألعاب الفيدي

التوزيع العام للخطة الدراسية

نوع المادة	عدد الساعات
(OB) إجباري	1500
(OP) إختياري	0
(PR) الممارسات الخارجية	0
(TFM) مشروع تخرج الماجيستير	0
الإجمالي 1500	

التوزيع العام للخطة الدراسية

الدورة	المادة	الطريقة	عدد الساعات
1°	أساسيات البرمجة	إجباري	150
1°	بنية البيانات والخوارزميات	إجباري	150
1°	البرمجة الموجهة للكائنات	إجباري	150
1°	وحدات تحكم وأجهزة ألعاب الفيديو	إجباري	150
1°	هندسة البرمجيات	إجباري	150
1°	محركات ألعاب الفيديو	إجباري	150
1°	أنظمة ذكية	إجباري	150
1°	البرمجة في الوقت الفعلي	إجباري	150
1°	تصميم وتطوير ألعاب الويب	إجباري	150
1°	الشبكات والأنظمة متعددة اللاعبين	إجباري	150

tech الجامعة التكنولوجية

أ. د./ Tere Guevara Navarro
رئيس الجامعة

*تصديق لاهاي أبوستيل. في حالة قيام الطالب بالتقدم للحصول على درجته العلمية الورقية وبتصديق لاهاي أبوستيل، ستتخذ مؤسسة TECH EDUCATION الإجراءات المناسبة لكي يحصل عليها وذلك بتكلفة إضافية.

الجامعة التيكنولوجية

ماجيستير خاص
برمجة ألعاب الفيدي

- طريقة التدريس: **أونلاين**
- مدة الدراسة: **12 أشهر**
- المؤهل العلمي من: **TECH الجامعة التكنولوجية**
- مواعيد الدراسة: **وفقًا لوتيرتك الخاصّة**
- الامتحانات: **أونلاين**

ماجيستير خاص
برمجة ألعاب الفيديو

tech الجامعة التيكنولوجية