

```
shot_date DES
```

```
as studio, COUNT(*) as count FROM
```

```
$studio_list->studio,"quick");
```

```
$studio_list->studio,"count" => $studio_list->count, "tit
```

```
udio");
```

```
ie('date not found');
```

```
image, image_date WHERE image_date.id=image.day_id AND image_date.shot_date='$sdate' AND image.enabled='y' AND i
```

```
ge_id);
```

**tech** الجامعة  
التكنولوجية

ماجستير خاص

برمجة مطورالواجهة الأمامية الكاملة  
(Front End Full Stack Developer)



الجامعة  
التكنولوجية **tech**

## ماجستير خاص برمجة مطورالواجهة الأمامية الكاملة (Front End Full Stack Developer)

« طريقة التدريس: عبر الإنترنت

« مدة الدراسة: 12 شهر

« المؤهل الجامعي من: TECH الجامعة التكنولوجية

« مواعيد الدراسة: وفقاً لوتيرتك الخاصة

« الامتحانات: عبر الإنترنت

رابط الدخول إلى الموقع الإلكتروني: [www.techtitute.com/ae/information-techngology/professional-master-degree/master-full-front-end-programming-stack-developer](http://www.techtitute.com/ae/information-techngology/professional-master-degree/master-full-front-end-programming-stack-developer)

# الفهرس

01	المقدمة	صفحة 4
02	الأهداف	صفحة 8
03	الكفاءات	صفحة 14
04	هيكل الإدارة وأعضاء هيئة تدريس الدورة التدريبية	صفحة 18
05	الهيكل والمحتوى	صفحة 22
06	المنهجية	صفحة 34
07	المؤهل العلمي	صفحة 42

# المقدمة

يعد مطورو الواجهة الأمامية Front End جزءًا لا غنى عنه في أي تطوير للويب. تتطلب طبيعتهم كحلقة وصل بين التصميم الجرافيكي وبرمجة الواجهة الخلفية Back End أن يكون لديهم معرفة محددة ومتطورة للغاية، والتي بدورها يجب أن تكون محدثة حتى يتمكنوا من مواجهة التحديات المتزايدة المتطلبات في الوقت الحاضر. نظراً لكونه تخصصاً يوفر فرصاً متعددة على المستوى المهني، فقد جمعت TECH أفضل فريق تدريس ممكن لتطوير برنامج جامعي شامل وكامل ومنهجي. بالاعتماد على الأدوات والمنهجيات الأكثر تقدماً في تطوير حلول للويب، تقدم هذه الشهادة الدافع اللازم لاتخاذ الخطوة نحو برمجة الواجهة الأمامية الكاملة Front End Full Stack.

كن جزءًا أساسيًا من أي تطوير للويب وقيادة وتنسيق فرق تكنولوجيا المعلومات بفضل منهجية العمل المتقنة والقيادة في هذا الماجستير الخاص"



يعد تطوير Full Stack خياراً مثيراً للاهتمام بشكل خاص لجميع محترفي تكنولوجيا المعلومات الذين يتطلعون إلى تعزيز حياتهم المهنية بشكل كبير. إن المهارات المطلوبة لتحقيق أداء جيد في هذا القطاع واسعة النطاق، مما يعني أن فرص الازدهار بل وقيادة فرق التطوير متعددة.

بفضل الشمولية التي تم بها تطوير جميع محتويات هذا البرنامج، سيتمكن الخريج من توجيه مسيرته المهنية نحو تطوير الواجهة الأمامية Front End للويب أو تخطيط الصفحات أو نحو أخصائي في تجربة العملاء أو DevOps. من خلال رؤية شاملة لكامل عملية بناء التطبيق/ الويب، سيتمكن عالم الكمبيوتر من التعامل مع أي نوع من المشاريع، كما سيوفر أيضاً تطوراً في أحدث التطورات في جميع عمليات دورة حياة البرنامج.

فرصة أكاديمية فريدة من نوعها للوصول إلى المعرفة التي تجمع بين أحدث النظريات الحاسوبية والممارسة المهنية عالية المستوى، والتي يقدمها فريق تدريس على أعلى مستوى من الجودة. إن خبرته على رأس العديد من المشاريع ذات الصلة في مجال الخدمات المصرفية الرقمية أو الاتصالات السلكية واللاسلكية تجعل المحتوى التعليمي أكثر ثراءً، حيث يقدم عدداً كبيراً من الحالات الحقيقية والقراءات التكميلية.

تعد المرونة الكبيرة لهذا التعليم من أبرز خصائصه الأخرى. لا توجد جداول زمنية محددة مسبقاً أو حصص دراسية حضورية، ويقرر الطالب متى وأين وكيف يأخذ عبء المقررات الدراسية بأكملها. يتوفر محتوى الفصل الدراسي الافتراضي بالكامل للتنزيل ويمكن دراسته من أي جهاز متصل بالإنترنت.

يحتوي هذا الماجستير الخاص في برمجة مطور الواجهة الأمامية الكاملة (Front End Full Stack Developer) على البرنامج التعليمي الأكثر اكتمالاً وتحديثاً في السوق. أبرز خصائصه هي:

- ♦ تطوير دراسات الحالة التي يقدمها خبراء في برمجة مطور الواجهة الأمامية الكاملة (Front End Full Stack Developer)
- ♦ محتوياتها البيانية والتخطيطية والعملية البارزة التي يتم تصورها بها تجمع المعلومات العلمية والعملية حول تلك التخصصات الأساسية للممارسة المهنية
- ♦ التمارين العملية حيث يمكن إجراء عملية التقييم الذاتي لتحسين التعلم
- ♦ تركيزها الخاص على المنهجيات المبتكرة
- ♦ دروس نظرية وأسئلة للخبير وعمل التفكير الفردي
- ♦ توفر المحتوى من أي جهاز ثابت أو محمول متصل بالإنترنت



قم بالتسجيل الآن ولا تفوّت الفرصة التي ستأخذك إلى الذروة في قيادة وتطوير أكثر مشاريع تكنولوجيا المعلومات طموحاً

سوف تعمق معرفتك بالمنهجيات المرنة وكيفية تطبيقها في عملية التطوير، مما يزيد من مهاراتك وكفاءاتك المستعرضة.

سيكون لديك إمكانية الوصول إلى مجموعة واسعة من المواد التعليمية، بدءاً من لغة برمجة Javascript إلى أدوات مثل CSS و Angular و ReactJS.



سوف تصل إلى مستوى متقدم من التخصص، بحيث تكون قادراً على بناء أي حل ويب مطلوب من منظور تجربة العملاء الحديثة وتتكيف مع السوق الحالية"

البرنامج يضم، في أعضاء هيئة تدريسه محترفين في القطاع، يجلبون إلى 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 تدريباً غامراً مبرمجاً للتدريب في حالات حقيقية.

يركز تصميم هذا البرنامج على التعلم القائم على حل المشكلات، والذي المهني في يجب أن تحاول من خلاله حل المواقف المختلفة للممارسة المهنية التي تنشأ من خلاله. للقيام بذلك، سيحصل على مساعدة من نظام فيديو تفاعلي مبتكر من قبل خبراء مشهورين.



# الأهداف

الهدف من درجة الماجستير الخاص في برمجة مطور الواجهة الأمامية الكاملة (Front End Full Stack Developer)، مع الأخذ في الاعتبار الخيارات المتعددة التي يوفرها التطوير، ليس سوى توفير المعرفة والتقنيات الأكثر تقدمًا في هذا المجال. بهذه الطريقة، يمكن لعالم الحاسوب البدء في تطوير مشاريعه الخاصة أو تعزيز مسيرته المهنية قبل إكمال المؤهل العلمي، وذلك بفضل النهج العملي البارز لجميع المحتويات المقدمة.

ستحقق أهدافك المهنية الأكثر طموحًا بفضل النهج  
المختلف لهذا التدريب، والذي سيرشدك خلال جميع مراحل  
تطوير الواجهة الأمامية الكاملة "Front End Full Stack"



## الأهداف العامة



- ♦ إنتاج المعرفة المتخصصة في الجوانب الرئيسية للبرمجة
- ♦ تشجيع التفكير بشكل خوارزمي
- ♦ توفير الأدوات والمهارات اللازمة للتطوير
- ♦ تعزيز اعتماد المنهجيات المرنة لتنفيذ المشروع
- ♦ تطوير المعرفة المتخصصة حول أساسيات الويب
- ♦ تشجيع استخدام أدوات وتقنيات تطوير الواجهة الأمامية Front End الحديثة
- ♦ تطوير تصميم ويب لمعالجة التخطيط بشكل صحيح
- ♦ تقييم المهارات المكتسبة



ستعمل على تحسين مهاراتك وكفاءاتك تدريجيًا، من خلال 10 وحدات دراسية تم إعدادها من خلال المعرفة الأكثر رسوخًا والخبرة التي اكتسبها جميع المعلمين"



### الوحدة 1. تطوير Full Stack Developer

- ♦ تطوير معارف متقدمة في البرمجة
- ♦ تشجيع استخدام أنظمة التحكم في الإصدار ومنصات استضافة التعليمات البرمجية
- ♦ تعزيز استخدام المنهجيات المرنة
- ♦ الخوض في المفاهيم الأساسية للإنترنت وعملها
- ♦ زيادة إتقان سطر الأوامر

### الوحدة 2. Front-end في البرمجة

- ♦ تحديد وفهم بناء الجملة الصحيح ل HTML و CSS
- ♦ استكشاف عناصر HTML المختلفة
- ♦ تحديد نهج التصميم التكييفي
- ♦ استخدام تنسيق العرض التقديمي لصفحات الويب من خلال تطبيق صحائف الأنماط المتتالية
- ♦ تضمين معالج CSS المسبق
- ♦ تحديد فوائد استخدام المعالج المسبق
- ♦ إنتاج المعرفة المتخصصة في أنظمة التصميم
- ♦ وضع معايير لاستخدام نظام التصميم

### الوحدة 3. لغة JAVASCRIPT المطبقة على Full Stack Developer

- ♦ تحديد الأنواع الأساسية والمعقدة التي تقدمها JavaScript
- ♦ تحليل الطرق المختلفة للبرمجة باستخدام اللغة واستخدامها بشكل صحيح في كل موقف
- ♦ تحديث المعرفة إلى أحدث الإصدارات
- ♦ اكتشاف البرمجة الوظيفية
- ♦ فحص البرمجة غير المتزامنة وخصائصها

### الوحدة 4. تخطيط الويب المطبق على Full Stack Developer

- ♦ تقييم تصميم الويب لمعرفة كيفية وضعه مؤقتاً
- ♦ فحص قواعد CSS الرئيسية
- ♦ تقديم منهجيات CSS المختلفة للحصول على تصميمات Responsive
- ♦ أساسيات مبادئ التطوير المتتالي ل CSS
- ♦ التعرف على تقنية Bootstrap في أي تصميم ويب
- ♦ تحليل مبادئ Bootstrap
- ♦ تطوير نموذج تصميم للويب باستخدام Bootstrap
- ♦ تحديد مبادئ التطوير في مشروع SaSS

### الوحدة 5. أدوات JavaScript. مكتبة ReactJs

- ♦ تحديد وظائف React
- ♦ إعداد مشروع باستخدام Create React App
- ♦ تحليل دورة حياة المكونات في React
- ♦ توليد معرفة متخصصة في وظائف React الحديثة مثل Hooks و Context
- ♦ تعيين الحالات العامة باستخدام Context
- ♦ إنشاء القوائم وعرضها وإنشاء النماذج باستخدام React
- ♦ تنفيذ التحقق من صحة المجالات في النماذج
- ♦ تصميم المكونات والعناصر
- ♦ تصحيح أخطاء تطبيقات React واختبارها ونشرها

### الوحدة 6. Framework de JavaScript Angular.

- ♦ تطوير المعرفة المتخصصة في بنية الإطار العملي Framework
- ♦ التعمق في منهجية Angular
- ♦ تحليل مفهوم المكونات
- ♦ تنظيم التعليمات البرمجية بشكل صحيح

### الوحدة 7. البرمجة بلغة NodeJS

- ♦ توليد معرفة متخصصة حول أنواع JavaScript ومشغليها
- ♦ تحليل أفضل طرق البرمجة باستخدام اللغة
- ♦ تحديث المعرفة إلى أحدث الإصدارات
- ♦ استكشاف البرمجة الوظيفية
- ♦ تطوير البرمجة غير المتزامنة ودوافعها
- ♦ اكتساب القدرة على إنشاء تطبيق باستخدام NodeJS

### الوحدة 8. قواعد البيانات لمطوري Full Stack

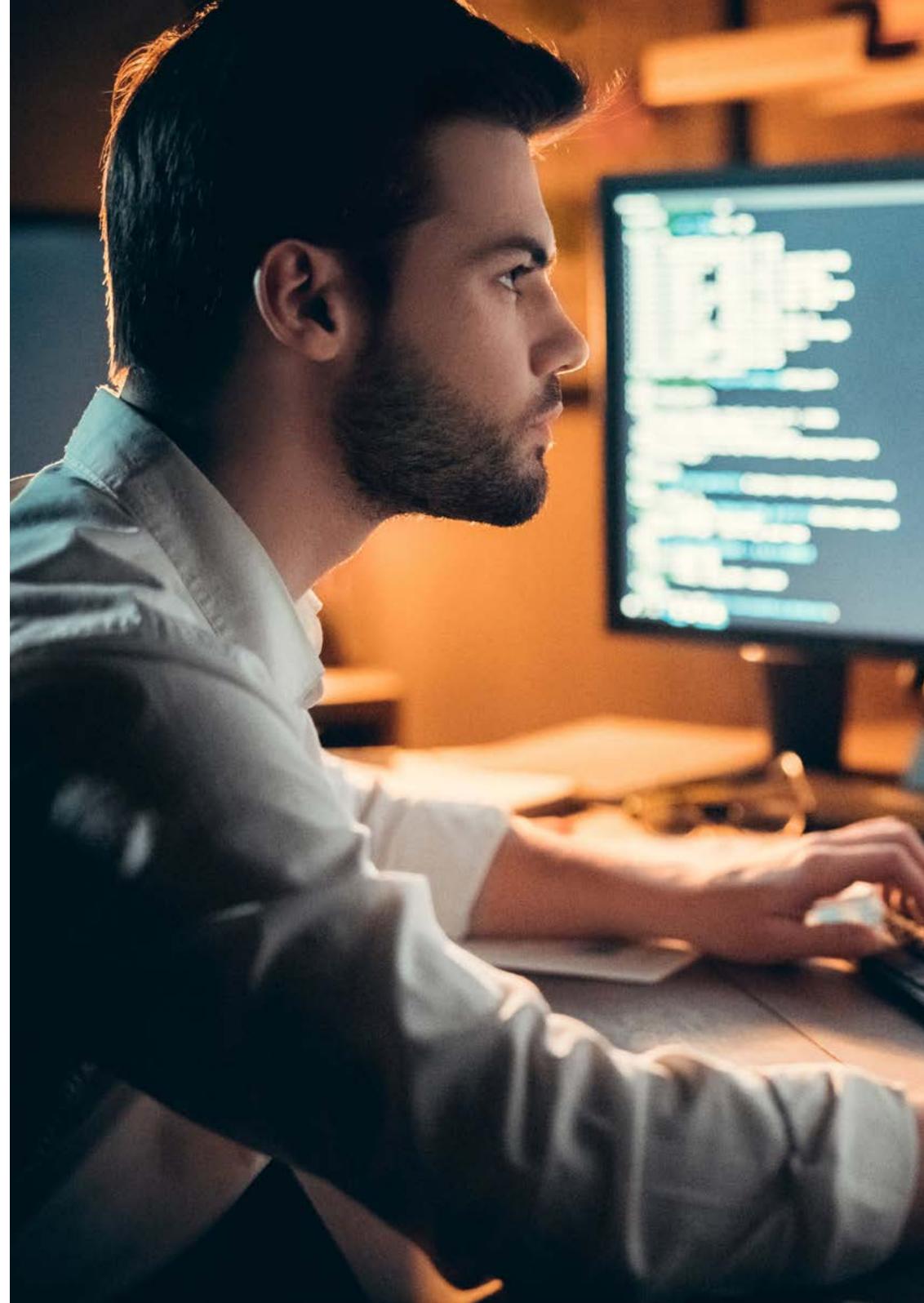
- ♦ تحديد سبب استخدام قاعدة بيانات في تطوير التطبيقات
- ♦ فحص أنواع قواعد البيانات المتاحة والاختلافات بينها
- ♦ تكوين فكرة واضحة عن الغرض من استخدام كل نوع من أنواع قواعد البيانات
- ♦ تحليل استخدام قاعدة البيانات في نماذج التطوير الحالية

### الوحدة 9. UX CX تجربة المستخدم و تجربة العميل تجربة العميل

- ♦ تحليل أهمية المستخدم في الوقت الحاضر والتعمق في ثقافة المراجعة
- ♦ تحقيق استراتيجيات القنوات المتعددة وإستراتيجيات التخصيص القائمة على التفاعلات الدقيقة
- ♦ دراسة تطور تحليلات الويب وصولاً إلى التحليلات السلوكية
- ♦ تحديد كيف ارتقى الذكاء الاصطناعي بتجربة العميل CX إلى المستوى التالي
- ♦ إنشاء أهم تقنيات تحليل تجربة الويب والتنقل وإمكانية الوصول
- ♦ تقديم منهجية التفكير التصميمي Design Thinking وعملية إنشاء تجربة المستخدم
- ♦ ابتكار أدوات ملموسة لوضع النماذج الأولية و WireFraming، بالإضافة إلى أطر العمل Frameworks لتطوير الواجهة الأمامية Front-End

### الوحدة 10. التكامل المستمر ونشر التطبيقات

- ♦ تحقيق فوائد اعتماد نموذج النشر التلقائي للتطبيقات
- ♦ تحديد الاختلافات بين التكامل المستمر والتسليم والنشر المستمرين
- ♦ تحديد الميزات الرئيسية لـ DevOps
- ♦ تقييم بعض الأدوات الرئيسية لتنفيذ خطوط أنابيب CI/CD
- ♦ تطوير العوامل الأساسية لتطوير التطبيقات الجاهزة لدعم عمليات CI/CD
- ♦ دراسة تقنيات الحاويات كركيزة أساسية لممارسة CI/CD

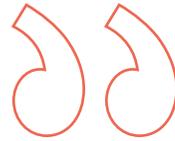


# الكفاءات

تتعدد المهارات التي يجب على مبرمج الواجهة الأمامية المتكاملة (Front End Full Stack Developer) وتطويرها، خاصةً في مجال يتسم بالتنافسية والتخصص الشديدين مثل تكنولوجيا المعلومات. لهذا السبب، يغطي المنهج الدراسي مختلف أنواع اللغات الأكثر شيوعًا في هذا المجال، بالإضافة إلى أدوات وفلسفة العمل التي يجب على عالم الحاسوب اتباعها لكي يبرز بتميز مهني. هذا ممكن بفضل الطبيعة متعددة التخصصات لفريق التدريس نفسه، الذي جمع كل ما لديه من معرفة فيما يتعلق بالمجالات المختلفة لتطوير الواجهة الأمامية Front End.



ستعطي سيرتك الذاتية دفعة عالية الجودة  
من خلال دمج العديد من المهارات والكفاءات  
المطلوبة في أعلى مستويات تطوير البرمجيات"





## الكفاءات العامة

- ♦ التعرف بشكل صحيح على بناء جملة لغات HTML و CSS
- ♦ تطوير معايير أفضل الممارسات لتطوير الويب
- ♦ توليد المعرفة المتخصصة بلغة JavaScript
- ♦ القدرة على تطوير أي نوع من التطبيقات باستخدام JavaScript
- ♦ تحليل مكتبة Bootstrap
- ♦ تنفيذ مشاريع التخطيط باستخدام SaSS Syntactically Awesome Stylesheets
- ♦ تحديد بناء جملة React وكيفية البرمجة باستخدامها
- ♦ تطبيق أفضل الممارسات على اللغة
- ♦ فحص عملية التحميل والوصول في كل نوع من أنواع قواعد البيانات الرائدة في منطقتك
- ♦ تقييم أهم الأدوات والتقنيات في تحليل تجربة العميل CX "مجموعة التقنيات stack المستخدمة عادةً في الشركات

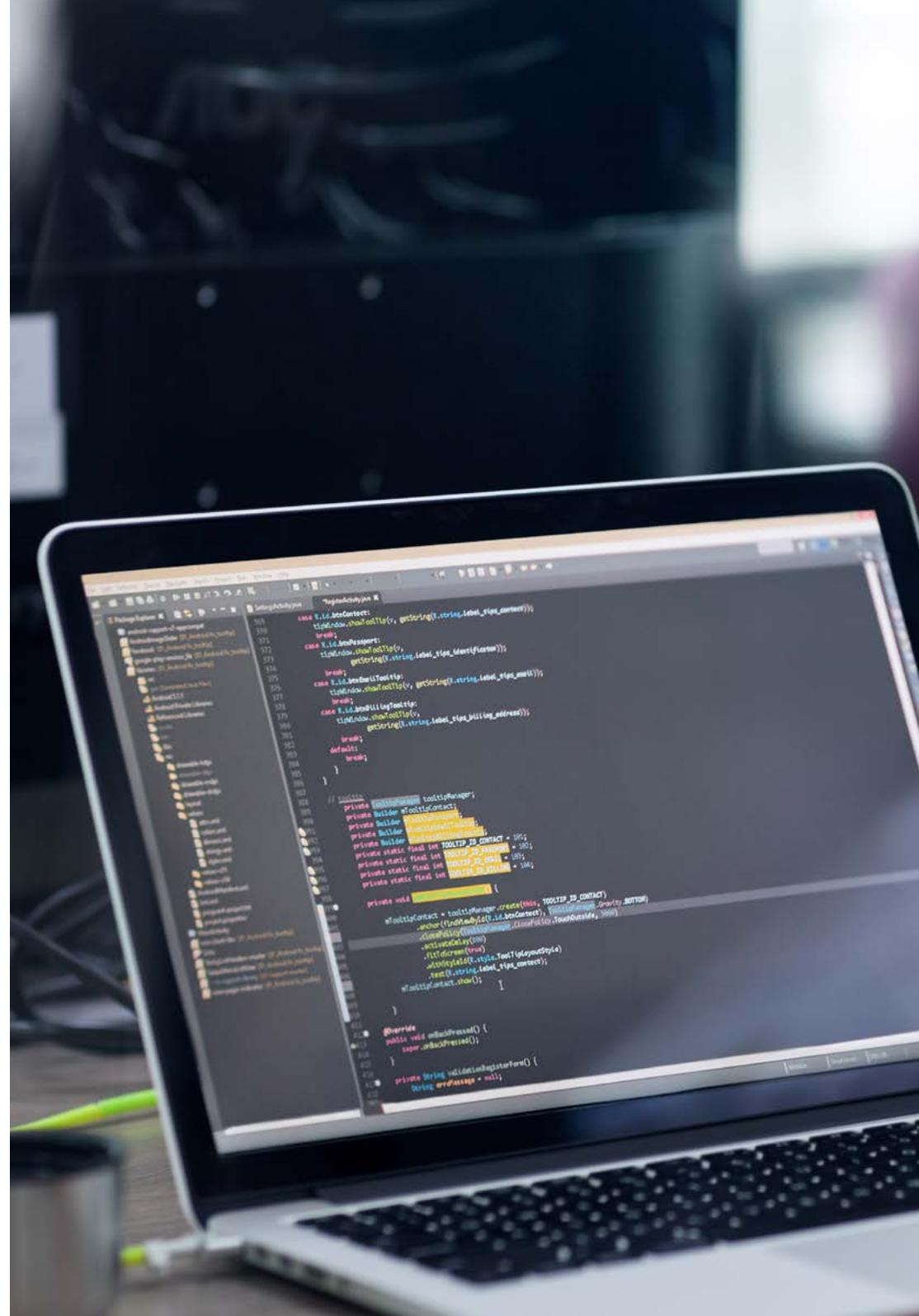
## الكفاءات المحددة



- ♦ تحليل هياكل البيانات المختلفة
- ♦ فحص تقنيات تصميم الخوارزميات وتفسيرها
- ♦ تحضير بيئة التطور
- ♦ استنساخ موقع إلكتروني
- ♦ إنشاء موقع إلكتروني باستخدام Bootstrap
- ♦ تجميع الترميز CSS باستخدام SaSS
- ♦ تطوير إطار عمل CSS Framework المستند إلى Bootstrap الخاص بك باستخدام SaSS
- ♦ إنشاء مشروع ووضعه موضع التنفيذ
- ♦ تحديد كيفية الاتصال وتحميل/استخراج البيانات من أنواع مختلفة من قواعد البيانات
- ♦ تحديد الممارسات وحالات الاستخدام والتقنيات والأدوات الخاصة بمنظومة CI/CD، الضرورية لدعم العملية برمتها



ستطوّر خلال الدورة التدريبية المهارات اللازمة للنجاح في تطوير الواجهة الأمامية (Front End) مما يعزز ليس فقط معرفتك الخاصة ولكن أيضًا كفاءاتك المستعرضة"



# هيكل الإدارة وأعضاء هيئة تدريس الدورة التدريبية

تولي TECH عناية خاصة إلى انتقاء أعضاء هيئة التدريس المسؤولين عن مؤهلاتها العلمية. إن الدقة التي يتم بها تشكيل الفرق لا تضمن فقط جودة المحتويات بنفسها، بل تضمن أيضاً مشاركة فريق ملتزم بالتطوير المهني لمحترفي تكنولوجيا المعلومات. بالتالي سيجد الطالب مواد تعليمية مصممة خصيصاً لتلبية احتياجاتهم الأكثر تطلباً، مع كل الدعم والمواد الإضافية المتوقعة من أكبر مؤسسة أكاديمية على الإنترنت في العالم.



هيكل الإدارة وأعضاء هيئة تدريس الدورة التدريبية | 19



سوف تكون قادرًا على الإجابة عن أسئلتك مباشرةً مع أعضاء هيئة التدريس، والحصول على برنامج تعليمي مخصص ويتناسب مع احتياجاتك"

## هيكـل الإدارة

### أ. Olalla Bonal, Martín

- ♦ متخصص تقني في Blockchain في IBM SPGI
- ♦ متخصص في المبيعات التقنية في IBM Blockchain.
- ♦ مدير الهندسة المعمارية. Blocknitive
- ♦ تقني إلكترونيات رقمية
- ♦ Infrastructure Architect -IT Blockchain Architect - مدير مشاريع تكنولوجيا المعلومات. مجالات العمل: البرمجيات والبنية التحتية والاتصالات السلكية واللاسلكية



## الأستاذة

### أ. Calzada Martínez, Jesús

- ♦ مهندس برمجيات أول في Devo
- ♦ مطور الواجهة المتكاملة Full Stack في Blocknitive
- ♦ مسؤول عن الواجهة الأمامية Front End في Infinia
- ♦ مطور الواجهة المتكاملة Full Stack في Resem
- ♦ مطور جافا في Hitec
- ♦ بكالوريوس في هندسة الكمبيوتر

### أ. Guerrero Díaz-Pintado, Arturo

- ♦ مهندس تقني لما قبل المبيعات عبر WatsonCustomer Engagement portfolio (حلول التسويق وتجربة العملاء Customer Experience) في إسبانيا والبرتغال واليونان وإسرائيل في شركة IBM
- ♦ مهندس شبكات البحث والتطوير في شركة Telefónica
- ♦ استشاري خدمات مهنية يعمل مع مؤسسات رائدة في أوروبا والشرق الأوسط وأمريكا اللاتينية من شركة IBM
- ♦ بكالوريوس في الهندسة العليا للاتصالات السلكية واللاسلكية من جامعة Alcalá و Danish Technical University
- ♦ التعاون المتميز في الجامعات ومراكز التعليم العالي الشهيرة في الموضوعات المتعلقة بالتكنولوجيا مثل الذكاء الاصطناعي و Internet of Things, Cloud, Customer Experience y Digital Transformation

#### أ. Frias Favero, Pedro Luis

- ♦ CTO - Swearit Technologies
  - ♦ COO - Key Identification
  - ♦ خبير في blockchain والتطبيقات اللامركزية - جامعة Alcalá
  - ♦ Full Stack Developer - Ironhack
  - ♦ مهندس صناعي متخرج من جامعة Yacambu
- #### أ. Gómez Rodríguez, Antonio
- ♦ مهندس الحلول السحابية Oracle soluciones Cloud في Oracle
  - ♦ مدير مشاريع Sopra Group
  - ♦ مدير مشاريع Everis
  - ♦ مدير المشاريع في الشركة العامة لإدارة البرامج الثقافية. مجلس الثقافة في الأندلس
  - ♦ محلل أنظمة المعلومات. Sopra Group
  - ♦ بكالوريوس في الهندسة العليا للاتصالات السلكية واللاسلكية من جامعة Politécnica في كاتالونيا
  - ♦ دراسات عليا في تكنولوجيا ونظم المعلومات من المعهد الكاتالوني للتكنولوجيا
  - ♦ E-Business Master، كلية La Salle لإدارة الأعمال

#### أ. Pintado San Claudio, Bruno

- ♦ منسق التطوير في IDAVinci
- ♦ مطور Java في مكتبة إسبانيا الوطنية
- ♦ مطور الدعم وتقني دعم الشبكات N1 في Sanitas
- ♦ تقني دعم الأنظمة في مجلس مدينة Alcobendas
- ♦ تقني اتصالات N1 لدى شركة ADIF في مركز Atocha للاتصالات السلكية واللاسلكية
- ♦ بكالوريوس في الهندسة التقنية للاتصالات السلكية واللاسلكية، مع تخصص أنظمة إلكترونية، جامعة Politécnica في مدريد
- ♦ بكالوريوس في هندسة إلكترونيات الاتصالات من جامعة Politécnica في مدريد

#### أ. Reyes Oliva, Luis

- ♦ مطور ومهندس سحابي في شركة IBM
- ♦ المدير التقني للعملاء لحسابات BBVA المتكاملة في IBM
- ♦ Cloud Executive Selling في IBM
- ♦ مهندس السحابة Cloud و DevOps في IBM
- ♦ مهندس برمجيات العملاء في Telefónica
- ♦ مهندس الحلول التقنية لشركة Rational
- ♦ Software Engineering Manager في Borland
- ♦ مدير تطوير البرمجيات وضمان الجودة في Altana Consulting
- ♦ بكالوريوس في الهندسة الحاسوبية من جامعة سالامانكا ألبابوية في مدريد

# الهيكل والمحتوى

بالنظر إلى أن الجهد والاستثمار الذي يبذله عالم الحاسوب في اختيار برنامج بهذه الخصائص ليس بالقليل، فإن TECH ملتزمة بمنهجية تدريس من الدرجة الأولى، أثبتت كفاءتها وربحياتها. تسمح منهجة إعادة التعلم (المعروفة بـ Relearning) للطلاب باكتساب أهم معارف البرنامج خلال المنهج الدراسي بأكمله، وبالتالي تجنب وقت الدراسة الثمين الذي يمكن أن يقضيه الطالب في المواد التكميلية العديدة المصاحبة لكل موضوع مقترح.

استكشف الوحدات والموضوعات المعرفية المختلفة من خلال ملخصات الفيديو ومقاطع الفيديو التفصيلية ومقاطع الفيديو التحفيزية التي أنشأها المعلمون بأنفسهم"



## الوحدة 1. تطوير Full Stack Developer

- 1.1 تطوير Full Stack Developer .1 البرمجة واللغات
  - 1.1.1 البرمجة
  - 2.1.1 أدوار البرمجة
  - 3.1.1 اللغات وإطار العمل Framework
  - 4.1.1 الخوارزمية
  - 5.1.1 خصائص الخوارزمية
- 2.1 مطور 2 Full Stack Developer الأنماط
  - 1.2.1 المتغيرات والثوابت
  - 2.2.1 الأنواع
  - 3.2.1 العاملين
  - 4.2.1 التصریحات
  - 5.2.1 الحلقات
  - 6.2.1 المهام والأهداف
- 3.1 هياكل البيانات في التطور
  - 1.3.1 أنواع الهياكل الخطية
  - 2.3.1 أنواع الهياكل الوظيفية
  - 3.3.1 أنواع هياكل الأشجار
- 4.1 تصميم الخوارزميات وتفسيرها
  - 1.4.1 التوازي في التنمية، فَرَّقْ تسد
  - 2.4.1 الخوارزميات الجشعة
  - 3.4.1 البرمجة الديناميكية
- 5.1 البيئة والأدوات اللازمة للتطوير الموجه ل Full Stack Developer
  - 1.5.1 إعداد البيئة لنظام التشغيل Mac OS
  - 2.5.1 إعداد البيئة من أجل Linux
  - 3.5.1 إعداد البيئة من أجل Windows

- 6.1 سطر الأوامر، النمط والأداء
  - 1.6.1 المحطة النهائية
  - 2.6.1 أجهزة المحاكاة
  - 3.6.1 مترجم الأوامر
  - 4.6.1 الأوامر الأولى
  - 5.6.1 البحث
  - 6.6.1 إدارة الملفات والمجلدات باستخدام واجهة سطر الأوامر
  - 7.6.1 Secure Shell, SSH
  - 8.6.1 الأوامر المتقدمة
- 7.1 Git مستودع البرامج
  - 1.7.1 مستودع البرامج Git
  - 2.7.1 استخدام Git
  - 3.7.1 مستودعات البرامج
  - 4.7.1 الفروع
  - 5.7.1 دورة العمل
  - 6.7.1 الأوامر
- 8.1 خدمة استضافة إصدارات الترميز
  - 1.8.1 خدمة استضافة إصدارات الترميز
  - 2.8.1 الموردون
  - 3.8.1 المستودعات
- 9.1 الانترنت
  - 1.9.1 الانترنت
  - 2.9.1 البروتوكولات المستخدمة في WWW
  - 3.9.1 بروتوكول HTTP
- 10.1 منهجيات مطور Full Stack
  - 1.10.1 Scrum
  - 2.10.1 XP
  - 3.10.1 Design Sprint

- 6.2 تحديد المواقع
  - 1.6.2 التموضع الثابت
  - 2.6.2 التموضع النسبي
  - 3.6.2 التموضع المطلق
  - 4.6.2 تحديد المواقع الثابتة
  - 5.6.2 Floats
- 7.2 التصميم المتكيف
  - 1.7.2 Viewport
  - 2.7.2 Media Queries
  - 3.7.2 وحدات CSS
  - 4.7.2 Images
  - 5.7.2 Frameworks
- 8.2 Layout حديث
  - 1.8.2 Flex
  - 2.8.2 Grid
  - 3.8.2 Flex مقابل Grid
- 9.2 المعالج المسبق
  - 1.9.2 Sass
  - 2.9.2 المتغيرات
  - 3.9.2 Mixins
  - 4.9.2 Bucles
  - 5.9.2 الدوال
- 10.2 نظام التصميم
  - 1.10.2 Bootstrap
  - 2.10.2 شبكة Bootstrap
  - 3.10.2 Footerg Header لموقعنا
  - 4.10.2 نماذج
  - 5.10.2 Cards
  - 6.10.2 القواعد

## الوحدة 2. Front-end في البرمجة

- 1.2 اللغة HTML
  - 1.1.2 مستند HTML
  - 2.1.2 عنصر Head
  - 3.1.2 عنصر Body
  - 4.1.2 النص
  - 5.1.2 الروابط التشعبية
  - 6.1.2 الصور
  - 7.1.2 Primer Site
- 2.2 اللغة HTML Layouts
  - 1.2.2 اللغة HTML العوامل
  - 2.2.2 Layout تقليدي
  - 3.2.2 Layout الدلالي
- 3.2 أوراق الأنماط المتتالية CSS Cascading Style Sheets
  - 1.3.2 تضمين CSS في مستند HTML
  - 2.3.2 تعليقات
  - 3.3.2 المحددات
  - 4.3.2 المحددات المتقدمة
- 4.2 خصائص أوراق الأنماط المتتالية CSS Cascading Style Sheets
  - 1.4.2 اللون
  - 2.4.2 النص
  - 3.4.2 الفئات الزائفة
  - 4.4.2 انتقالات
  - 5.4.2 الرسوم المتحركة
  - 6.4.2 الرسوم المتحركة للعناصر
  - 7.4.2 الرسوم المتحركة المتقدمة
- 5.2 نموذج الصندوق
  - 1.5.2 الطول والعرض
  - 2.5.2 الهامش
  - 3.5.2 الحشوة

## الوحدة 3. لغة JAVASCRIPT المطبقة على Full Stack Developer

- 5.3 هياكل البيانات: مصفوفة Array
  - 1.5.3 هيكـل البيانات. مصفوفة
  - 2.5.3 مصفوفة Array الأنماط
  - 3.5.3 مصفوفات Arrays متداخلة
  - 4.5.3 طرق المصفوفة Array
- 6.3 POO: النموذج الأولي Prototype والفئات
  - 1.6.3 POO: البرمجة الموجهة للأشياء
  - 2.6.3 النماذج الأولية
  - 3.6.3 الفئات
  - 4.6.3 البيانات الخاصة
  - 5.6.3 الفئات الفرعية
  - 6.6.3 Apply و Call
- 7.3 أنواع JavaScript
  - 1.7.3 Set
  - 2.7.3 WeakSet
  - 3.7.3 Map
  - 4.7.3 WeakMap
  - 5.7.3 التعبيرات العادية
- 8.3 استخدامات JavaScript
  - 1.8.3 Date
  - 2.8.3 Math
  - 3.8.3 Symbol
  - 4.8.3 JSON
- 9.3 JavaScript في Browser
  - 1.9.3 تضمين JavaScript في موقع ويب
  - 2.9.3 DOM
  - 3.9.3 الفعاليات
  - 4.9.3 Storage في المتصفح
- 10.3 البرمجة غير المتزامنة
  - 1.10.3 البرمجة غير المتزامنة
  - 2.10.3 Event Loop
  - 3.10.3 Calbacks
  - 4.10.3 Promises
  - 5.10.3 Async/Await

- 1.3 الأنواع والعوامل البدائية
  - 1.1.3 لغة JavaScript
  - 2.1.3 الأعداد ومشغليها
  - 3.1.3 السلاسل النصية ومشغلاتها
  - 4.1.3 القيم المنطقية
  - 5.1.3 التحويل بين الأنواع
- 2.3 وحدات التحكم في التدفق والهيكل
  - 1.2.3 التعبيرات والجمل
  - 2.2.3 المتغيرات والثوابت
  - 3.2.3 If كانت الجملة
  - 4.2.3 جمل For, While
- 3.3 الدوال
  - 1.3.3 الدوال
  - 2.3.3 المعايير
  - 3.3.3 الدوال كإعدادات
  - 4.3.3 تحليل المتغيرات
  - 5.3.3 النطاقات Scopes المتداخلة
  - 6.3.3 Hoisting
  - 7.3.3 Closures
  - 8.3.3 العودية
- 4.3 هياكل البيانات: الأهداف
  - 1.4.3 نوع Object الشبيئ
  - 2.4.3 إحداث الأشياء
  - 3.4.3 الدخول إلى قيم الشيء
  - 4.4.3 إضافة خصائص أو إزالتها
  - 5.4.3 أشياء متداخلة
  - 6.4.3 إعادة هيكلة Destructuring الأشياء
  - 7.4.3 طرق من نوع الشبيئ Object
  - 8.4.3 Spread Operator
  - 9.4.3 الثبات

- 6.4 . Sass
  - 1.6.4 . البرمجيات كخدمة (SaSS)
  - 2.6.4 . تركيب SaSS
  - 3.6.4 . تشغيل وتجميع SaSS
  - 4.6.4 . هيكلية دليل SaSS
  - 7.4 . استخدام SaSS
    - 1.7.4 . المتغيرات في Sass
    - 2.7.4 . نمذجة مشروعنا
    - 3.7.4 . بناء الجملة ل SaSS
    - 8.4 . منطق SaSS
      - 1.8.4 . Mixins
      - 2.8.4 . Maps
      - 3.8.4 . الوظائف وهيكل التحكم
    - 9.4 . التخطيط باستخدام Bootstrap
      - 1.9.4 . Bootstrap
      - 2.9.4 . Layout de Bootstrap
      - 3.9.4 . نماذج
      - 4.9.4 . Box Model باستخدام Bootstrap
      - 5.9.4 . الألوان والخطوط
      - 6.9.4 . الروابط والأزرار
      - 7.9.4 . إظهار العناصر وإخفاؤها باستخدام Bootstrap
      - 8.9.4 . Flex Box باستخدام Bootstrap
      - 9.9.4 . المكونات
    - 10.4 . Theming Bootstrap
      - 1.10.4 . الكتابة فوق Bootstrap باستخدام SaSS Software as a Service
      - 2.10.4 . بنية الملفات
      - 3.10.4 . إنشاء إطارنا للعمل CSS Framework الخاص بنا CSS Cascading Style Sheets

## الوحدة 4. تخطيط الويب المطبق على Full Stack Developer

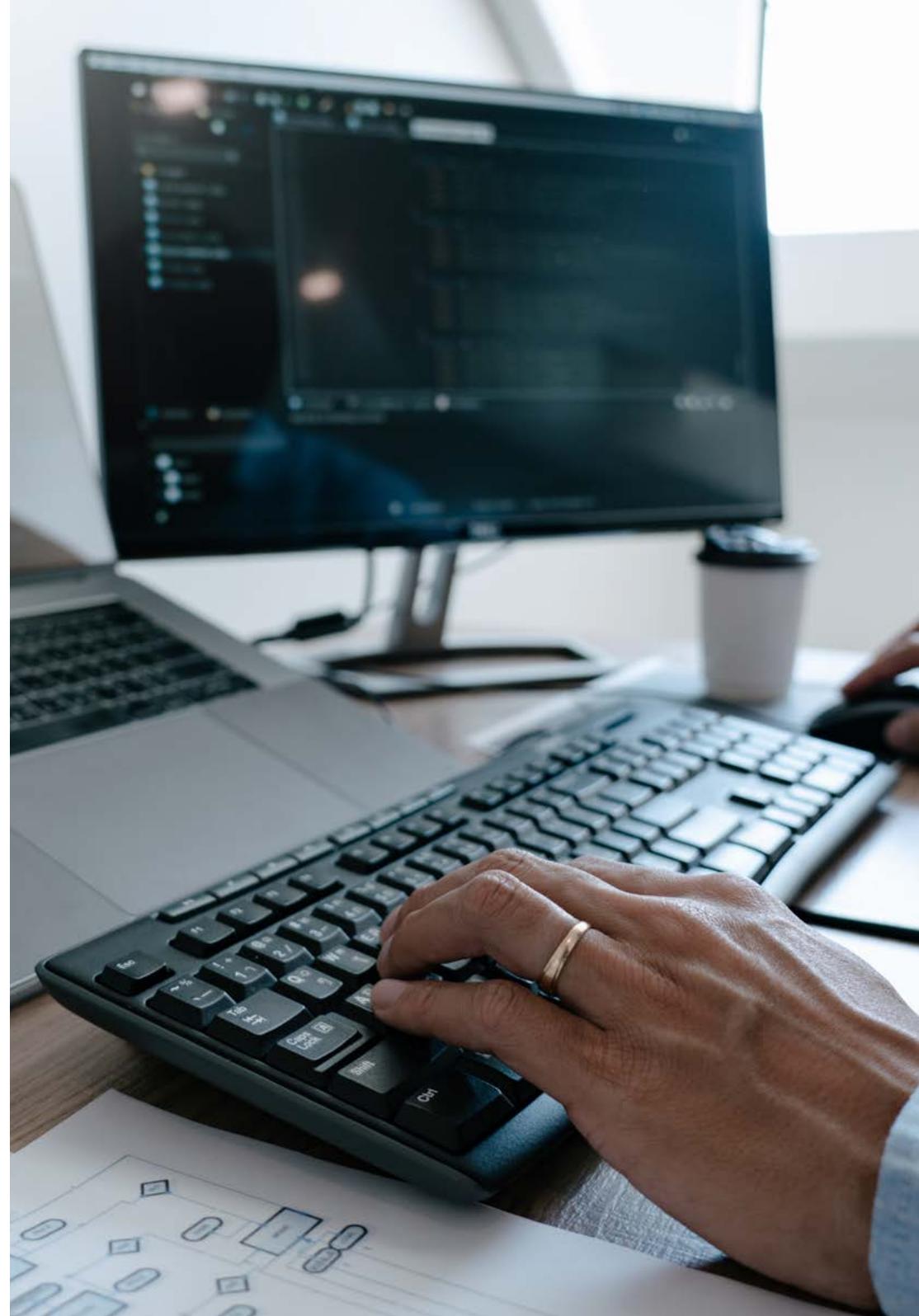
- 1.4 . CSS والتخطيط
  - 1.1.4 . التخطيط بالجدول
  - 2.1.4 . التصميم السائل
  - 3.1.4 . عصر Responsive
  - 4.1.4 . Mobile First مقابل Desktop First
- 2.4 . CSS وقواعد تصميم الويب
  - 1.2.4 . محددات
  - 2.2.4 . الفئات الزائفة
  - 3.2.4 . العناصر الزائفة
- 3.4 . تخطيط باستخدام CSS
  - 1.3.4 . قواعد Box Model
  - 2.3.4 . الطباعة
  - 3.3.4 . الألوان
  - 4.3.4 . صور
  - 5.3.4 . الأموال
  - 6.3.4 . الجداول
  - 7.3.4 . نماذج
  - 8.3.4 . إظهار العناصر وإخفاؤها
  - 9.3.4 . متغيرات CSS
- 4.4 . تصميم Responsive وتصميم سلس
  - 1.4.4 . العناصر العائمة
  - 2.4.4 . Grid CSS
  - 3.4.4 . Media Queries
  - 4.4.4 . Flex Box
- 5.4 . سلسلة CSS
  - 1.5.4 . أولوية قواعد CSS
  - 2.5.4 . الكتابة فوق القواعد
  - 3.5.4 . الفئات مقابل. المعرّفات

## الوحدة 5. أدوات JavaScript. مكتبة ReactJs

- 1.5 أداة ReactJS Javascript
- 1.1.5 أداة ReactJS
- 2.1.5 Create React App
- 3.1.5 JavaScript Syntax Extension
- 2.5 مكونات ReactJS
- 1.2.5 المكونات
- 2.2.5 Props
- 3.2.5 التصيير
- 3.5 الفعاليات في مكتبة ReactJS
- 1.3.5 إدارة الفعاليات
- 2.3.5 إدارة الفعاليات عبر الإنترنت
- 3.3.5 الفعاليات في مكتبة ReactJS
- 4.5 تكوين خطافات ReactJS Hook
- 1.4.5 حالة المكوّن
- 2.4.5 خطافات Hook الحالة
- 3.4.5 خطافات Hook الأثر
- 4.4.5 Custom Hooks
- 5.4.5 آخرون Hooks
- 5.5 ReactJS في Component Context
- 1.5.5 ReactJS في Component Context
- 2.5.5 استخدام Context
- 3.5.5 هيكل Context
- 4.5.5 React. Create Context
- 5.5.5 Context. Provider
- 6.5.5 Class. Context Type
- 7.5.5 Context. Consumer
- 8.5.5 Context.displayName
- 9.5.5 Context للتطبيق العملي لاستخدام Context
- 6.5 التوجيه في ReactJs
- 1.6.5 Router
- 2.6.5 React Router
- 3.6.5 التركيب
- 4.6.5 التوجيه الأساسي
- 5.6.5 التوجيه الديناميكي
- 6.6.5 المكونات الأولية
- 7.6.5 React Router Hooks
- 7.5 استخدام القوائم والنماذج باستخدام ReactJS
- 1.7.5 القوائم والخطافات
- 2.7.5 النماذج وعمليات التحقق
- 3.7.5 React Hook Forms
- 8.5 استخدام الأنماط في ReactJS
- 1.8.5 تصميم تقليدي
- 2.8.5 التصميم عبر الإنترنت
- 3.8.5 إضافة مكتبة نظام التصميم
- 9.5 إجراء اختبارات في Javascript الأدوات
- 1.9.5 Testing
- 2.9.5 Jest JavaScript Testing Framework
- 3.9.5 Visual Testing and Documentation
- 10.5 نشر التعليمات البرمجية باستخدام ReactJS
- 1.10.5 Hosting
- 2.10.5 الموردون
- 3.10.5 إعداد المشروع
- 4.10.5 النشر على Heroku

## الوحدة 6. إطار العمل JavaScript Angular في Framework

- 1.6. إطار العمل Framework ذو الزاوية وهندسته
  - 1.1.6. واجهة سطر الأوامر في الزاوية Angular
  - 2.1.6. البنيات
  - 3.1.6. مساحة العمل Workspace والهيكل
  - 4.1.6. المحيط
- 2.6. مكونات إطار العمل Framework الزاوي
  - 1.2.6. دورة الحياة
  - 2.2.6. تغليف البصر
  - 3.2.6. التفاعلات بين المكونات
  - 4.2.6. إسقاط المحتوى
- 3.6. مكونات إطار العمل Framework الزاوي
  - 1.3.6. استيفاء النص
  - 2.3.6. تصريحات
  - 3.3.6. Property Binding
  - 4.3.6. Attribute Binding و Class, Style
  - 5.3.6. Two-Way Binding و Event Binding
  - 6.3.6. Pipes
- 4.6. مديري إطار العمل Framework الزاوي
  - 1.4.6. التوجيهات الزاوية
  - 2.4.6. توجيهات السمّة
  - 3.4.6. التوجيهات الهيكلية
- 5.6. الخدمات وحقن التبعيات
  - 1.5.6. الخدمات
  - 2.5.6. حقن التبعيات
  - 3.5.6. Service Providers



## الوحدة 7. البرمجة بلغة NodeJS

- .1.7 Node.js وهندستها
- .1.1.7 NPM وإدارة الحزم
- .2.1.7 تنفيذ البرنامج
- .3.1.7 الوحدات النمطية
- .4.1.7 إنشاء وحدة نمطية
- .5.1.7 Loop حلقة الأحداث
- .2.7 خادم الواجهة الخلفية و HTTP و Express و Sockets
- .1.2.7 وحدة HTTP النمطية
- .2.2.7 Express
- .3.2.7 Socket.io
- .3.7 قواعد البيانات و Caché
- .1.3.7 MongoDB
- .2.3.7 Mongoose
- .3.3.7 SQL
- .4.3.7 Sequelize
- .5.3.7 Redis
- .4.7 نظام الملفات ونظام التشغيل
- .1.4.7 وحدة نظام الملفات
- .2.4.7 Os Module
- .3.4.7 Cluster module
- .5.7 الفعاليات و Buffers و Streams
- .1.5.7 الفعاليات
- .2.5.7 Buffers
- .3.5.7 Streams
- .6.7 Testing
- .1.6.7 Jest
- .2.6.7 Mocha
- .3.6.7 TDD - Cucumber

- .6.6 Routing والتصفح
- .1.6.6 التطبيق مع Routing
- .2.6.6 التوجيه الأساسي
- .3.6.6 المسارات المتداخلة
- .4.6.6 المعايير
- .5.6.6 الوصول والتفويض
- .6.6.6 Lazy Loading للوحدات النمطية
- .7.6 RxJS
- .1.7.6 المراقبون
- .2.7.6 Observers
- .3.7.6 الاشتراكات
- .4.7.6 العاملين
- .8.6 النماذج و HTTP
- .1.8.6 الأشكال التفاعلية
- .2.8.6 التحقق من الميادين
- .3.8.6 النماذج الديناميكية
- .4.8.6 الالتماسات
- .5.8.6 Interceptors
- .6.8.6 الأمان
- .9.6 الرسوم المتحركة
- .1.9.6 التحولات المحفزات Triggers
- .2.9.6 انتقالات المسارات
- .3.9.6 الاختلافات بين التحولات
- .10.6 Testing en el Framework Angular
- .1.10.6 اختبار الخدمات
- .2.10.6 اختبار المكونات
- .3.10.6 اختبار التوجيهات و الأنابيب pipes

## الوحدة 8. قواعد البيانات لمطوري Full Stack

- 1.8 . قواعد البيانات لمطوري Full Stack
  - 1.1.8 . قاعدة البيانات ضمن تطوير التطبيقات
  - 2.1.8 . إمكانيات قواعد البيانات
  - 3.1.8 . SQL Structured Query Language
- 2.8 . انتقاء قاعدة البيانات
  - 1.2.8 . الطلب أو الخدمة التي سيتم النظر فيها
  - 2.2.8 . فئات قواعد البيانات
  - 3.2.8 . بانوراما قاعدة البيانات
- 3.8 . التطوير باستخدام MySQL
  - 1.3.8 . التطوير باستخدام MySQL
  - 2.3.8 . نشر النموذج العلائقي باستخدام MySQL
  - 3.3.8 . اتصال MySQL
- 4.8 . التطوير باستخدام Oracle Database
  - 1.4.8 . التطوير باستخدام Oracle DB
  - 2.4.8 . نشر النموذج
  - 3.4.8 . الاتصال ب Oracle Database
- 5.8 . التطوير باستخدام Oracle SQL Server
  - 1.5.8 . Oracle SQL Server
  - 2.5.8 . نشر النموذج
  - 3.5.8 . الاتصال SQL Server
- 6.8 . التطوير باستخدام NoSQL
  - 1.6.8 . مقارنة مع قواعد بيانات SQL
  - 2.6.8 . إنشاء قاعدة بيانات في MongoDB
  - 3.6.8 . الاتصال ب MongoDB
- 7.8 . التطوير باستخدام الشبكات
  - 1.7.8 . التطوير باستخدام الشبكات
  - 2.7.8 . إنشاء قاعدة بيانات باستخدام Neo4z
  - 3.7.8 . الاتصال باستخدام Neo4z

- 7.7 . الهندسة المعمارية وأفضل الممارسات
  - 1.7.7 . DRY
  - 2.7.7 . صلبة
  - 3.7.7 . CRUD
  - 4.7.7 . MVC
  - 5.7.7 . المتزامنة
  - 6.7.7 . الخدمات المصغرة
  - 7.7.7 . البنى سداسية الشكل
- 8.7 . نسخة مطبوعة
  - 1.8.7 . الأنواع والواجهات والفئات
  - 2.8.7 . الوظائف والوحدات النمطية
  - 3.8.7 . العامة
  - 4.8.7 . Namespaces
  - 5.8.7 . مصممو الديقور
  - 9.7 . API REST
    - 1.9.7 . Get
    - 2.9.7 . Post
    - 3.9.7 . Put
    - 4.9.7 . Delete
    - 5.9.7 . Swagger
  - 6.9.7 . بناء واجهة برمجة تطبيقات REST API باستخدام Express
- 10.7 . إنشاء تطبيق واحتواءه باستخدام NestJS
  - 1.10.7 . Nest CLI
  - 2.10.7 . Docker
  - 3.10.7 . إنشاء تطبيق

- 4.9. تحليلات المستخدم
  - 1.4.9. هيكل الموقع الإلكتروني
  - 2.4.9. تحليلات المستخدم
  - 3.4.9. تحليلات المستخدم المتقدمة
- 5.9. نيلسن وتأثيرها على تجربة العملاء CX
  - 1.5.9. نيلسن وأثره على CX
  - 2.5.9. تقنيات اختبار المستخدم User Testing
- 6.9. أدوات تجربة العملاء Customer Experience
  - 1.6.9. أدوات متطورة
  - 2.6.9. إمكانية التنقل
  - 3.6.9. امكانيه الوصول
- 7.9. منهجيات جديدة
  - 1.7.9. تحدي المستخدم
  - 2.7.9. عملية تجربة المستخدم UX
  - 3.7.9. بحث المستخدم
- 8.9. توصيل التصميم
  - 1.8.9. التخطيط الشبكي
  - 2.8.9. تصميم أدوات الاتصال
  - 3.8.9. أدوات اتصال التصميم المتقدمة
- 9.9. تصميم واجهة المستخدم UI
  - 1.9.9. تصميم واجهة المستخدم UI
  - 2.9.9. واجهات الويب والهاتف المحمول
  - 3.9.9. مكونات الويب والمحمول
- 10.9. إعداد CX
  - 1.10.9. إعداد CX
  - 2.10.9. تصميم تجارب جديدة
  - 3.10.9. واجهات

- 8.8. التطوير باستخدام قاعدة البيانات ذات القيمة الرئيسية
  - 1.8.8. التطوير باستخدام قاعدة البيانات k-v
  - 2.8.8. إنشاء قاعدة بيانات باستخدام Redis
  - 3.8.8. الاتصال باستخدام Redis
- 9.8. قواعد البيانات مع أنواع البيانات الأخرى
  - 1.9.8. Elastic Search
  - 2.9.8. Inmemory Database
  - 3.9.8. التطوير باستخدام البيانات المكانية
- 10.8. قواعد بيانات. الجوانب المتقدمة
  - 1.10.8. قواعد البيانات في التطورات السحابية الأصلية cloud native
  - 2.10.8. قواعد البيانات في هندسة الخدمات المصغرة
  - 3.10.8. CI/CDD وقواعد البيانات

## الوحدة 9. CX UX تجربة المستخدم و تجربة العميل تجربة العملاء

- 1.9. Customer Experience (تجربة الزبون)
  - 1.1.9. Customer Experience (تجربة الزبون) (CX)
  - 2.1.9. احتياجات المستهلكين الجديدة
  - 3.1.9. Feedback في Customer Experience
- 2.9. التقنيات المبتكرة
  - 1.2.9. الآلات التي تفكر
  - 2.2.9. طرق جديدة لمشاركة المعلومات
  - 3.2.9. قياس ما لا يمكن قياسه
- 3.9. قنوات تفاعل المستخدم
  - 1.3.9. تحليل العملاء
  - 2.3.9. إضفاء الطابع الشخصي
  - 3.3.9. قنوات متعددة لتفاعل المستخدم

- .8.10 CI/CD باستخدام Jenkins. المراجع
- .1.8.10 CI/CD باستخدام Jenkins
- .2.8.10 Pipelines Jenkins
- .3.8.10 أفضل الممارسات باستخدام Jenkins
- .9.10 النظام البيئي في CI/CD
- .1.9.10 تنظيم النظام البيئي
- .2.9.10 ادوات متطورة
- .3.9.10 Dagger المستقبل
- .10.10 المراحل الأخيرة من دورة البرمجيات الموجهة نحو CI/CDD
- .1.10.10 تطبيق الذكاء الاصطناعي AI على عملية CI/CD
- .2.10.10 DevSecOps
- .3.10.10 Chaos Engineering



سيكون لديك تحت تصرفك أحدث الموارد التعليمية، مع إمكانية الوصول المجاني إلى الفصل الافتراضي على مدار 24 ساعة في اليوم"

## الوحدة 10. التكامل المستمر ونشر التطبيقات

- .1.10 التكامل المستمر والنشر المستمر CI/CD
- .1.1.10 استخدام الذكاء المستمر والنشر المستمر (CI/CD)
- .2.1.10 الاختلافات بين التكامل المستمر والنشر المستمر (CI/CD)
- .3.1.10 التكامل المستمر والنشر المستمر. فوائد CI/CDD
- .2.10 نماذج التنمية الجديدة
- .1.2.10 من Waterfall إلى DevOps
- .2.2.10 دليل الأسلوب: العوامل الـ 12
- .3.2.10 Cloud Native، والخدمات المصغرة و Serverless
- .3.10 DevOps، ما بعد CI/CD
- .1.3.10 DevOps
- .2.3.10 DevOps Continuous Everything.
- .3.3.10 DevOps vs SRE
- .4.10 تقنية الحاويات 1 - Docker
- .1.4.10 الحاويات. المساهمات
- .2.4.10 Docker. البنيات
- .3.4.10 عملية النشر باستخدام Docker
- .5.10 تقنية الحاويات 2 - Kubernetes
- .1.5.10 التنسيق
- .2.5.10 Kubernetes
- .3.5.10 النظام البيئي في Kubernetes
- .6.10 تهيئة البنية التحتية باستخدام GitOps
- .1.6.10 بنية تحتية غير قابلة للتغيير
- .2.6.10 GitOps
- .3.6.10 أدوات GitOps
- .7.10 أنبوب البيانات Pipelines والأتمتة. حالات استخدام CI/CD
- .1.7.10 التكامل المستمر
- .2.7.10 النشر والتسليم المستمر
- .3.7.10 عمليات التحقق التلقائي
- .4.7.10 الممارسات الجيدة في CI/CDD

# 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.

## 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 يرسى 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 الحاسبات 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 المحاضرة الجامعية، سيواجه الطلاب عدة حالات حقيقية. يجب عليهم دمج كل معارفهم والتحقيق والجدال والدفاع عن أفكارهم وقراراتهم.



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



## منهجية إعادة التعلم (Relearning)

تجمع جامعة TECH بين منهج دراسة الحالة ونظام التعلم عن بعد، 100% عبر الانترنت والقائم على التكرار، حيث تجمع بين 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 طبيعة التعليم العالمي، يسمى *Relearning* أو إعادة التعلم.

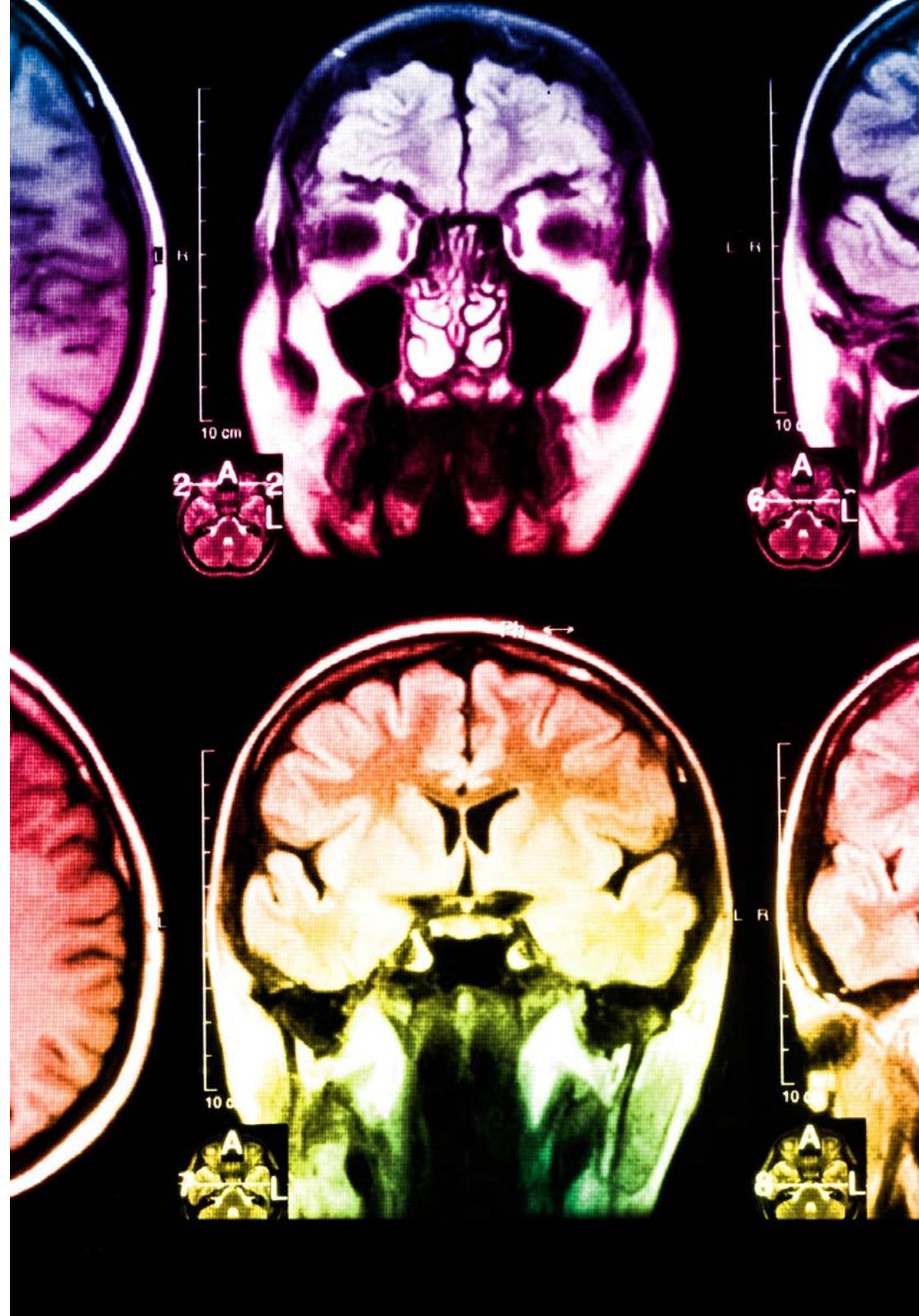
جامعتنا هي الجامعة الوحيدة الناطقة باللغة الإسبانية المصممة لهذا المنهج الناجح. في عام 2019، تمكنا من تحسين مستويات الرضا العام لطلابنا من حيث (جودة التدريس، جودة المواد، هيكل الدورة، الأهداف...) فيما يتعلق بمؤشرات أفضل جامعة عبر الإنترنت باللغة الإسبانية.

في برنامجنا، التعلم ليس عملية خطية، ولكنه يحدث في شكل لولبي (نتعلم 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 الحصين بالبحر، لكي نحفظ 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 المُعدَّة بعناية للمهنيين:

### المواد الدراسية



يتم إنشاء جميع محتويات التدريس من قبل المتخصصين الذين سيقومون بتدريس البرنامج الجامعي، وتحديداً من أجله، بحيث يكون التطوير التعليمي محدداً وملموساً حقاً.

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 للطلاب.

### 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

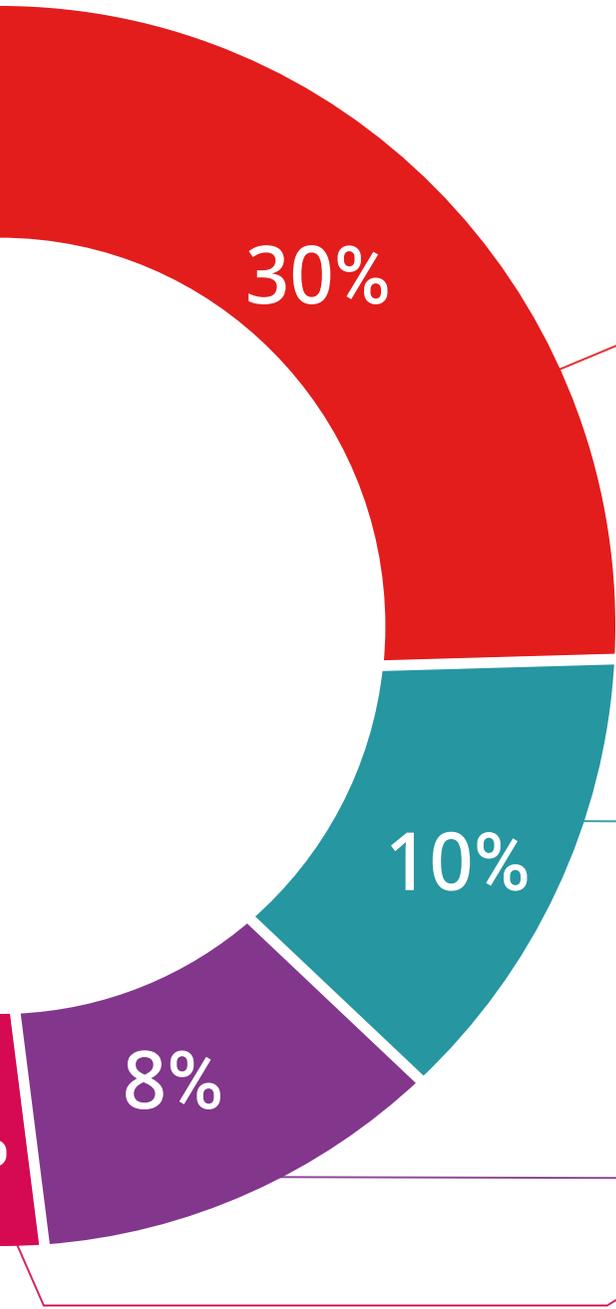


سيقومون بتنفيذ أنشطة لتطوير مهارات وقدرات محددة في كل مجال مواضيعي. التدريب العملي والديناميكيات لاكتساب وتطوير المهارات والقدرات التي يحتاجها المتخصص لنموه في إطار العولمة التي نعيشها.

### 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.



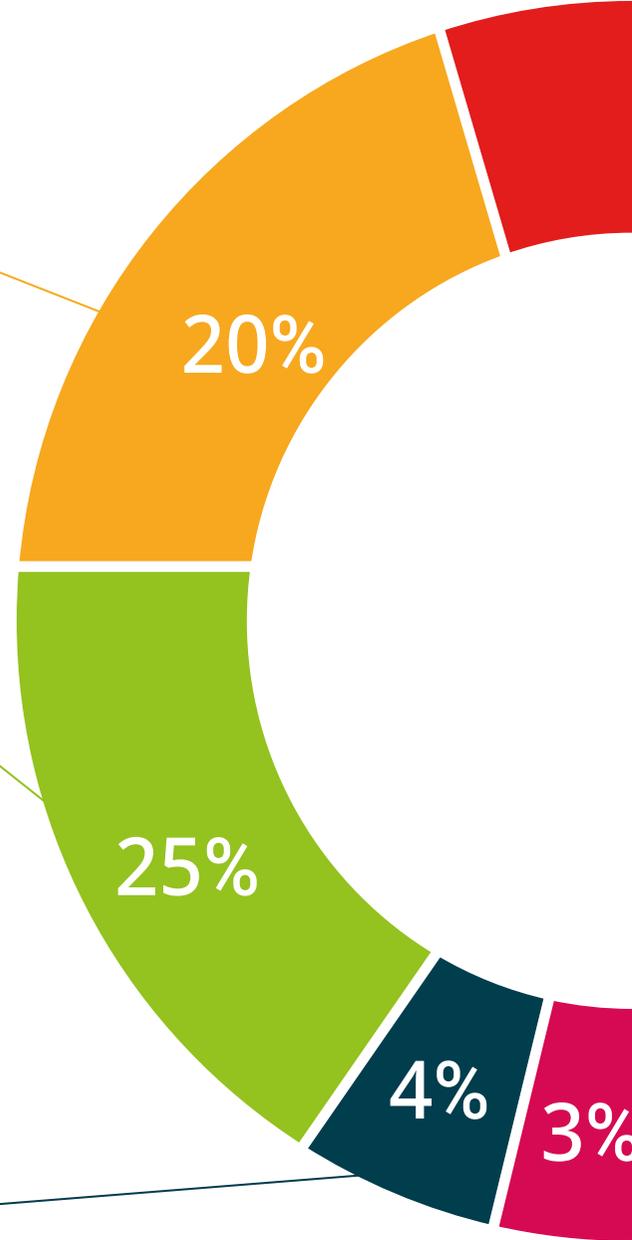
#### ملخصات تفاعلية

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 مايكروسوفت 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.



# المؤهل العلمي

يضمن الماجستير الخاص في برمجة مطورالواجهة الأمامية الكاملة (Front End Full Stack Developer) بالإضافة إلى التدريب الأكثر دقة وحداثة، الحصول على مؤهل الماجستير الخاص الصادر عن TECH الجامعة التكنولوجية.



اجتاز هذا البرنامج بنجاح واحصل على مؤهلك العلمي الجامعي  
دون الحاجة إلى السفر أو القيام بأية إجراءات مرهقة"



إن المؤهل الصادر عن **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: ماجستير خاص في برمجة مطور الواجهة الأمامية الكاملة (Front End Full Stack Developer)

طريقة الدراسة: عبر الإنترنت

مدة الدراسة: 12 شهر

تحتوي درجة الماجستير الخاص في برمجة مطور الواجهة الأمامية الكاملة (Front End Full Stack Developer) على البرنامج العلمي الأكثر اكتمالا وحداثة في السوق.

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

**محاضرة جامعية في برمجة مطور الواجهة الأمامية الكاملة (Front End Full Stack Developer)**

التوزيع العام للخطة الدراسية			التوزيع العام للخطة الدراسية		
الدرجة	المادة	الدرجة	عدد الساعات	نوع المادة	عدد الساعات
1*	تطوير Full Stack Developer	1*	1500	إلزامي	0
1*	Front-end في الواجهة	1*	0	إلزامي	0
1*	تطوير الواجهة الأمامية على Full Stack Developer	1*	0	الممارسات الخارجية (PP)	0
1*	أدوات JavaScript مثل: React	1*	0	مشروع تخرج الماجستير (TFM)	1500
1*	إطار عمل AngularJS	1*	0		
1*	البرمجة بلغة NodeJS	1*	0		
1*	قواعد البيانات لمطوري Full Stack	1*	0		
1*	API REST للمستهلك وخدمة العميل تحية العمل	1*	0		
1*	الذكاء الاصطناعي ونظم التطبيقات	1*	0		

**tech** | الجامعة التكنولوجية

*Tere Guevara*  
أ.د. / Tere Guevara Navarro  
رئيس الجامعة

**tech** | الجامعة التكنولوجية

**شهادة تخرج**  
هذه الشهادة منوطة إلى

..... مع وثيقة تحقيق شخصية رقم .....  
المواطن/المواطنة .....  
لاجتيازها/لاجتيازها بنجاح والحصول على برنامج

ماجستير خاص  
في  
برمجة مطور الواجهة الأمامية الكاملة (Front End Full Stack Developer)

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

*Tere Guevara*  
أ.د. / Tere Guevara Navarro  
رئيس الجامعة

TECH: AFW0238 | technote.com/certificates | لأكواد التبريد الخاص بالجامعة

الجامعة  
التكنولوجية  
**tech**

ماجستير خاص  
برمجة مطور الواجهة الأمامية الكاملة  
(Front End Full Stack Developer)

« طريقة التدريس: عبر الإنترنت

« مدة الدراسة: 12 شهر

« المؤهل الجامعي من: TECH الجامعة التكنولوجية

« مواعيد الدراسة: وفقاً لوتيرتك الخاصة

« الامتحانات: عبر الإنترنت

# ماجستير خاص برمجة مطورالواجهة الأمامية الكاملة (Front End Full Stack Developer)