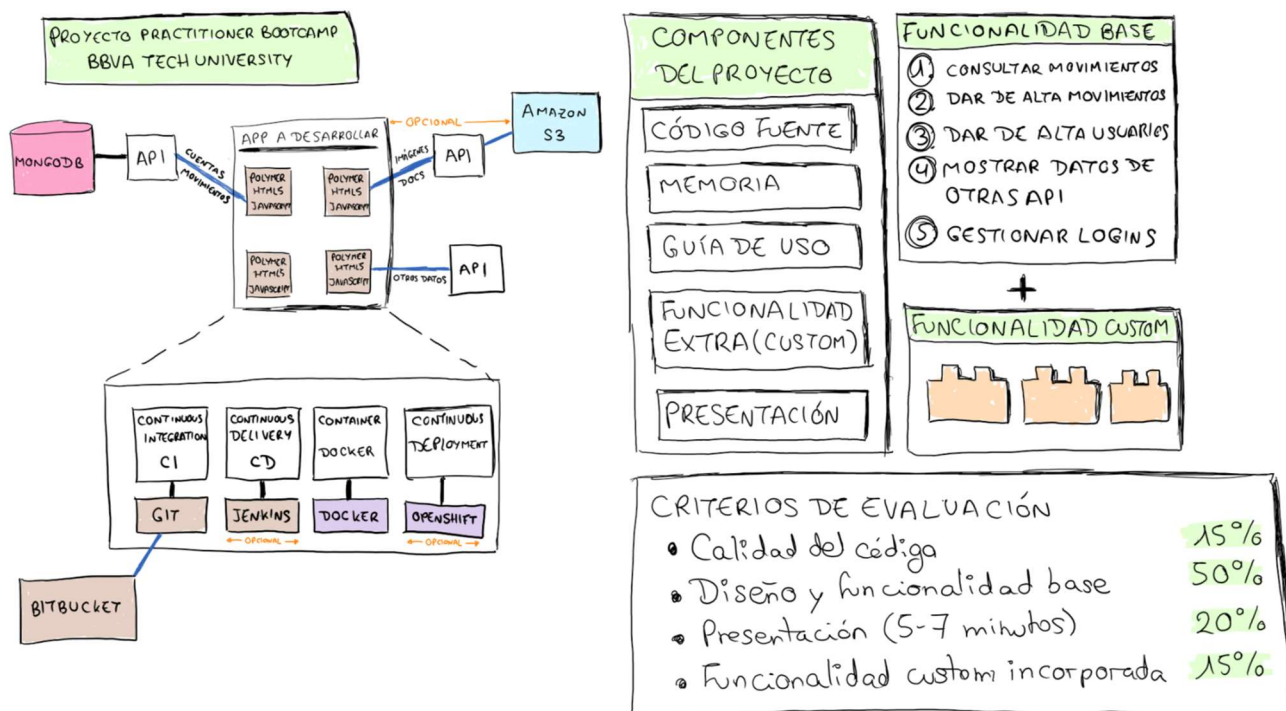


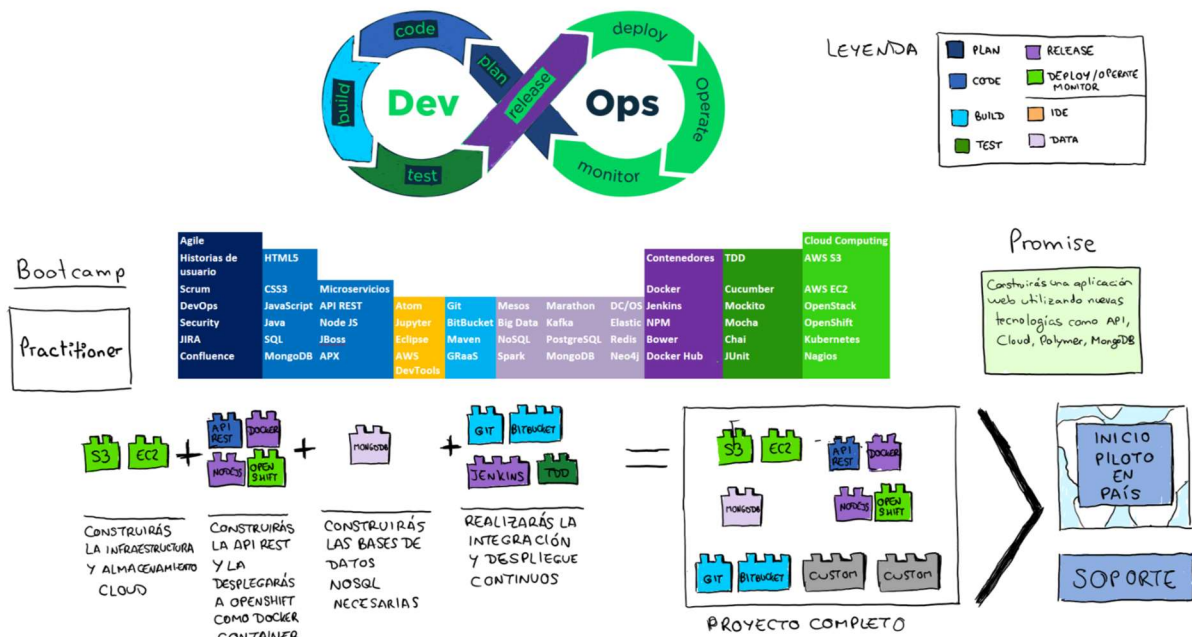
Proyecto BBVA Tech University 2019 - Perú (2ª Edición)

El presente documento tiene que objetivo describir la arquitectura y las tareas a realizar para completar el proyecto final del programa BBVA Tech University.

Arquitectura del proyecto



Visual Journey



Ejercicios para realizar en el aula - piezas del proyecto Practitioner

Al comienzo del Practitioner Bootcamp el tutor facilitará el esqueleto del proyecto sobre el que irán construyendo las diferentes piezas / ejercicios indicados a continuación. Dicho esqueleto permitirá hacer un seguimiento continuo de los cambios introducidos y evitará problemas de integración posteriores.

MK-DT-M2-E1 MongoDB

- **Objetivos:** Crear su propia base de datos MongoDB y colecciones en MongoDB local y en MLab
- **Definición del ejercicio:** El ejercicio consistirá en:
 - Crear una colección de movimientos de cuentas y una de usuarios.
 - Los documentos de la colección de cuentas y movimientos tendrán los siguientes "campos":
 - IdCuenta (número)
 - Cliente (número)
 - Listado de movimientos:
 - Id (autonumérico)
 - Fecha (date)
 - Importe (número)
 - Tipo (texto)

- Los documentos de la colección de usuarios tendrán los siguientes “campos”:
 - IdCliente (autonumérico)
 - Nombre (texto)
 - Apellido (texto)
 - Email (texto)
 - Password (texto)
- **Entorno:** Para realizar el ejercicio los alumnos utilizarán un navegador web para el uso de MLab y el servidor MongoDB instalado en la máquina virtual de la formación más el cliente 3T Studio.
- **Forma de entrega:** El ejercicio se entregará directamente al faculty. El faculty enviará la calificación de cada alumno al tutor para actualizar el Excel de tutor.
- **Evaluación:** La calificación final vendrá condicionada a cubrir los requisitos de las colecciones definidas en los detalles del ejercicio y a que funcionen las queries básicas con la API de MLab y con 3T Studio
- **Fechas de realización:** El ejercicio se realizará durante las sesiones de MongoDB.

MK-DV-M2-E1 SDLC – Proyecto final

- **Objetivos:** Integrar todas las piezas del proyecto en el aula
- **Definición del ejercicio:** El ejercicio consistirá en integrar con las tecnologías vistas en clase y con los siguientes criterios iniciales:
 - La aplicación deberá consumir servicios REST de MLab
 - La aplicación consumirá usuarios, cuentas y movimientos de cuentas disponibles en MongoDB (MLab)
 - La aplicación deberá funcionar dentro de un contenedor Docker
 - El código fuente estará disponible en repositorios BitBucket o GitHub
- En la definición del proyecto se plantearán los siguientes elementos:
 - Trabajo a realizar
 - Sprints
 - Arquitectura de la app
- Los criterios adicionales a incluir en el proyecto serán:
 - Crear componentes Polymer adicionales o HTML5 que permitan ampliar la funcionalidad de la app. Por ejemplo: alta de movimientos, generación de informes de gastos, visualización de datos, integración con otras API, etc.
- El objetivo es conseguir desarrollar dicho proyecto y entregarlo como proyecto final de BBVA Tech University
- **Entorno:** Para realizar el ejercicio los alumnos utilizarán un navegador web con acceso a BitBucket.
- **Forma de entrega:** El ejercicio se entregará directamente al faculty. El faculty enviará la calificación de cada alumno al tutor para actualizar el Excel de tutor.
- **Evaluación:** La calificación final vendrá condicionada a los entregables identificados.
- **Fechas de realización:** El ejercicio se realizará durante la primera semana del Practitioner Bootcamp

MK-DV-M2-E2 DevOps

- **Objetivos:** Integrar las diferentes piezas de DevOps a utilizar en el proyecto (Git, BitBucket y Docker)
- **Definición del ejercicio:** En este taller se utilizarán todas las herramientas vistas en las sesiones formativas que permitan construir un pipeline DevOps.
Estas herramientas son:
 - Git
 - BitBucket
 - Docker
- **Entorno:** Para realizar el ejercicio los alumnos utilizarán las herramientas Git, Jenkins y Docker disponibles en la máquina virtual y un navegador con acceso a la cuenta BitBucket.
- **Forma de entrega:** El ejercicio se entregará directamente al faculty. El faculty enviará la calificación de cada alumno al tutor para actualizar el Excel de tutor.
- **Evaluación:** La calificación final vendrá condicionada a conseguir completar las fases de publicar con Git en BitBucket, crear un pipeline en Jenkins que cree el contenedor Docker y verificar que el contenedor se puede iniciar y funciona correctamente.
- **Fechas de realización:** El ejercicio se realizará durante la sesión de DevOps.

MK-DV-M3-E1 Polymer

- **Objetivos:** Construir componentes Polymer que permitan interactuar con el Back-End
- **Definición del ejercicio:** El ejercicio consistirá en la creación de componentes Polymer que servirán para ofrecer la funcionalidad del proyecto.
 - El primero de los componentes servirá para iniciar sesión en la cuenta bancaria del usuario a través del servicio REST MLab y contendrá dos cajas de texto para indicar usuario y password además de un botón que servirá para solicitar el acceso. La petición se hará a través de una llamada a un servicio REST que nos indicará OK o KO.
 - El segundo de los componentes permitirá consultar los movimientos de la cuenta bancaria del usuario. Tendrá un desplegable para elegir la cuenta, otro para el tipo de movimiento (retirada de efectivo, factura, nómina, etc) y un botón para invocar la consulta. También se podrá consultar por fecha por lo que habrá dos selectores de fecha para elegir el rango de búsqueda. La consulta se realizará vía GET.
 - Ambas peticiones se enviarán a la API MLab y posteriormente a la API NodeJS personalizada.
- **Entorno:** Para realizar el ejercicio los alumnos utilizarán las herramientas NodeJS, npm y polymer-cli disponibles en la máquina virtual. Como editor de código utilizarán Atom o Visual Studio Code disponibles también en la máquina virtual
- **Forma de entrega:** El ejercicio se entregará directamente al faculty. El faculty enviará la calificación de cada alumno al tutor para actualizar el Excel de tutor.

- **Evaluación:** La calificación final vendrá condicionada a crear los dos componentes e integrarlos en la aplicación SPA.
- **Fechas de realización:** El ejercicio se realizará durante las sesiones de Polymer.

MK-IA-M1-E1 Amazon S3 (OPCIONAL)

- **Objetivos:** Conocer el funcionamiento del servicio de almacenamiento S3 de Amazon Web Services (AWS)
- **Definición del ejercicio:** El ejercicio consistirá en crear un bucket (público) de Amazon S3 y almacenar en él un fichero de texto con las siguientes características:
 - Nombre del archivo: alumno.txt
 - Contenido del archivo: *Tu nombre completo*
- **Entorno:** Para realizar el ejercicio los alumnos únicamente necesitarán acceso al portal de AWS a través de un navegador web. Al comienzo del Practitioner se enviará a los alumnos sus datos de acceso.
- **Forma de entrega:** El ejercicio se entregará directamente al faculty indicando el dominio completo asignado en AWS para acceder al fichero. El faculty enviará la calificación de cada alumno al tutor para actualizar el Excel de tutor.
- **Evaluación:** La calificación final vendrá condicionada a poder acceder correctamente al fichero identificado a través de la URL de AWS
- **Fechas de realización:** El ejercicio se realizará durante las sesiones de AWS S3.

MK-IA-M1-E2 Amazon EC2 (OPCIONAL)

- **Objetivos:** Conocer el funcionamiento del servicio de computación EC2 de Amazon Web Services (AWS)
- **Definición del ejercicio:** El ejercicio consistirá en crear una instancia de AMI con las siguientes características:
 - Sistema operativo: Ubuntu
 - Acceso SSH: puerto 22
 - Tamaño de instancia: t2.micro
- **Entorno:** Para realizar el ejercicio los alumnos únicamente necesitarán acceso al portal de AWS a través de un navegador web. Al comienzo del Practitioner se enviará a los alumnos sus datos de acceso.
- **Forma de entrega:** El ejercicio se entregará directamente al faculty indicando el dominio completo asignado en AWS para acceder a la máquina virtual. El faculty enviará la calificación de cada alumno al tutor para actualizar el Excel de tutor.
- **Evaluación:** La calificación final vendrá condicionada a poder acceder correctamente a la máquina virtual identificada a través de la URL de AWS.
- **Fechas de realización:** El ejercicio se realizará durante las sesiones de AWS EC2.

MK-PA-M1-E1 REST-RESTful

- **Objetivos:** Dar los primeros pasos en la creación de una API REST utilizando NodeJS.
- **Definición del ejercicio:** El ejercicio consistirá en un taller de creación de consumo y creación de diferentes servicios REST disponibles utilizando Node.js y Express.
- **Entorno:** Para realizar el ejercicio los alumnos utilizarán las herramientas NodeJS y NPM disponibles en la máquina virtual. Como editor de código utilizarán Atom o Visual Studio Code también disponibles en la máquina virtual.
- **Forma de entrega:** El ejercicio se entregará directamente al faculty indicando el enlace al repositorio BitBucket. El faculty enviará la calificación de cada alumno al tutor para actualizar el Excel de tutor.
- **Evaluación:** La calificación final vendrá condicionada a la creación correcta de las peticiones GET, PUT, POST y DELETE en la API NodeJS/Express.
- **Fechas de realización:** El ejercicio se realizará durante las sesiones de NodeJS.

MK-PA-M1-E2 Docker

- **Objetivos:**
- **Definición del ejercicio:** El ejercicio consistirá en trabajar con los comandos de Docker para crear, modificar y desplegar contenedores.
 - Para ello habrá que seguir los siguientes pasos:
 - Obtener una imagen Docker pública
 - Revisar los componentes y el Dockerfile
 - Lanzar el contenedor y comprobar que funciona
 - La segunda parte consistirá en crear un contenedor Docker con las siguientes características:
 - Aplicación NodeJS
 - Expuesta en el puerto 3000
- **Entorno:** Para realizar el ejercicio los alumnos utilizarán la herramienta Docker disponible en la máquina virtual y Docker Hub como servicio online.
- **Forma de entrega:** El ejercicio se entregará directamente al faculty indicando el enlace al contenedor en Docker Hub. El faculty enviará la calificación de cada alumno al tutor para actualizar el Excel de tutor.
- **Evaluación:** La calificación final vendrá condicionada a la publicación del contenedor Docker y a la ejecución correcta del mismo.
- **Fechas de realización:** El ejercicio se realizará durante las sesiones de Docker.

De lunes a viernes se reservarán los últimos 30 minutos de clase para poder completar los ejercicios y resolver dudas en el aula además de poder ir avanzando en el proyecto.

Trabajo adicional personalizado

Los alumnos ampliarán la funcionalidad del proyecto añadiendo nuevos servicios REST/RESTful, componentes Polymer, diseños personalizados (HTML5, CSS3, JavaScript), elementos de seguridad, etc.

Como ejemplo de trabajo adicional personalizado se pueden plantear los siguientes elementos y funcionalidades:

- Nuevos componentes Polymer
- Consumo de servicios adicionales (API, REST, servicios web, ...)
- Funcionalidad adicional de la aplicación:
 - Gestión de usuarios
 - Gestión de movimientos
 - Gráficos de gastos, movimientos, cuentas, ...

Soporte y tutorización

Los alumnos podrán plantear sus dudas sobre el contenido o sobre los ejercicios a través de las siguientes vías:

- Correo electrónico del tutor: tutor.techuniversity.peru@bbva.com
- Canal Slack de la edición actual: bbvatechuper2019.slack.com
- Comunidad Google+ de BBVA Tech University
<https://plus.google.com/communities/116589760700423271483>
- Comunidades BBVA Google+ de disciplinas. Actualmente contamos con las siguientes:
 - APX: <https://plus.google.com/communities/116633944230895189316>
 - Cells: <https://plus.google.com/communities/109489428539696676001>
 - IaaS: <https://plus.google.com/communities/100817552588963929669>
 - DevOps & Advanced Engineering:
<https://plus.google.com/communities/101819433491864466825>
 - PaaS: <https://plus.google.com/communities/101393123615293428551>

Adicionalmente a estos canales de comunicación se organizarán sesiones de tutorías, en función de las necesidades, para dar soporte a los alumnos en sus diferentes sedes de manera presencial. Se convocarán dos sesiones de tutorías con posibilidad de acceso por Webex si el alumno prefiere seguirlo en remoto y evitar el desplazamiento.

Opción de proyecto en grupo

Se podrá realizar y presentar el proyecto en grupos de máximo 2 personas indicándolo durante la formación presencial al tutor.

Proceso de defensa del proyecto

El proyecto será defendido ante un Tribunal elegido por el equipo de BBVA Tech University. Para esta defensa es importante seguir los siguientes pasos:

1. Los alumnos facilitarán acceso al código en un repositorio BitBucket para que el Tutor y los miembros del Tribunal puedan revisarlo antes del día de la defensa. Dicho código tendrá que ser facilitado con la antelación a la fecha de la defensa indicada al final de este bloque.
2. Los alumnos subirán a la carpeta Google Drive creada para tal efecto los siguientes documentos:
 - a. Memoria del proyecto: documento PPT o similar
 - b. Guía de uso del proyecto: documento PPT o similar. Podrá combinarse con el documento de la memoria.
 - c. Enlace a la aplicación funcionando, salvo que se trate de un proyecto que se ejecute de manera local.
3. El día asignado para defender el proyecto el alumno contará con 5-7 minutos para realizar dicha defensa. Para ello seguirá los siguientes pasos:
 - a. Presentación del trabajo realizado y explicación de las funcionalidades utilizando la memoria y la guía de uso.
 - b. Demo funcional del proyecto
 - c. Preguntas y respuestas planteadas por el Tribunal

Aquellos alumnos que participen en el Practitioner Bootcamp entregarán los documentos y el acceso al repositorio BitBucket definidos en los puntos anteriores.

El **plazo de entrega del proyecto** será de **seis semanas** a partir de la fecha de finalización del Practitioner (último día presencial del bootcamp).

Las fechas para la **defensa del proyecto** se irán comunicando tras finalizar el bootcamp.

Happy RESTing!