ANEXO EXAMEN DE CERTIFICACIÓN

|  |  |
| --- | --- |
| Plan de estudio | Desarrollo de aplicaciones Fullstack JavaScript Trainee |
| Anexo | Caso Bootcamp Estudiantes Curso, día uno |

# Caso “Bootcamp Estudiantes Curso”

“Bootcamp Coders” es una academia dedicada a la entrega de servicios de capacitación orientados a la transformación digital, esto significa que se especializan principalmente en cursos de programación de sistemas web en diversos lenguajes de programación, como, por ejemplo, Fullstack Java, Fullstack Python, Fullstack JS, desarrollo Frontend con VUE, entre otros.

La academia tiene una existencia de aproximadamente 15 años en la industria de la capacitación, en la cual, aproximadamente hace 5 años se reestructuró para poder capacitar a personas bajo metodologías innovadoras denominadas bootcamp, activas y aula invertida.

Dada esta circunstancia, la demanda de capacitaciones creció considerablemente, gracias a esta actualización por requerimientos del mercado, por lo que, la academia se ve en la necesidad de crear un sistema web que les permita llevar el control de estudiantes, relatores, ayudantes y cursos, en dicho proyecto es en donde usted participará como desarrollador fullstack, junto a un equipo de profesionales con diversas competencias para poder implementar una solución apropiada según lo que esperan los gerentes de la organización.

El equipo se conformará por un profesional UX/UI, un desarrollador frontend, un líder de proyecto definido como un Scrum Master y usted que participará del proyecto como desarrollador fullstack.

El proyecto busca, como se mencionó previamente, generar un registro de estudiantes, cursos, relatores y ayudantes para poder generar un acceso rápido y persistente de los datos de los participantes de los diversos cursos y de los cursos propiamente tal. A continuación, se listan los requisitos funcionales de alto nivel:

* El sistema debe permitir consultar los cursos existentes, comuna y por consiguiente región asignada a dicho curso
* El sistema debe permitir la consulta de los tutores (ya sean ayudantes o relatores) que se encuentran participando de cada curso
* El sistema debe permitir identificar como se compone un curso, con respecto a los módulos que se impartirán dentro de este y los tutores que se harán cargo de cada uno

A la fecha, ya se ha avanzado en el proyecto y se cuenta con el siguiente avance:

* Ya se cuenta con un prototipo del aplicativo
* Existe un modelo de datos diseñado para dar solución a los requerimientos planteados hasta el momento
* Existe el script SQL que permite crear y poblar la base de datos, como aun no se define si se trabajará en MySQL o PostgreSQL, se encuentran los Script para ambas bases de datos, en el caso que se seleccione uno u otro, considerar las descargas de las librerías necesarias para poder realizar la conexión entre el ORM de Django con el motor de base de datos seleccionado

# Modelo de Datos

A continuación, se presenta el modelo de datos diseñado por el arquitecto en conjunto con un analista

Diagrama

Descripción generada automáticamente

Como se puede observar, en primera instancia se definen los planes formativos que se ejecutan dentro de la academia, estos están compuestos por diversos módulos que le dan forma a cada plan

Desde cada plan formativo, se puede entender que se crean cursos basados en cada uno de los planes formativos, los cuales tienen una fecha de inicio y termino, en base al hecho que se crean desde un plan formativo, es que también se componen de módulos que le dan forma a cada curso y el total de hora que durará el curso

Para cada curso y módulo, se define un relator y ayudante, los cuales se relacionan entre si mediante la tabla “Curso\_Modulo\_Tutor”, en donde, mediante las claves primarias se genera una relación entre un curso, un módulo y los tutores (relator y ayudante)

En última instancia, cada curso se asocia a una comuna, lo que no limita a que los estudiantes puedan ser de otras comunas, dado que, los cursos son 100% remotos, de la misma forma, cada estudiante y tutor viven dentro de una comuna y por consiguiente en una región

# Requerimientos a desarrollar

El Scrum Master, que es el encargado de definir las tareas dentro del equipo, le ha solicitado a usted que realice las siguientes tareas

1. Realizar consultas a la base de datos
2. Crear monitor de estudiantes
3. Crear API Rest que disponibilice la información del monitor de estudiantes

A continuación, se especifican con mayor detalle cada uno de los requerimientos:

1. Realizar consultas a la base de datos

Uno de los gerentes de la academia ha solicitado algunos reportes de la base de datos mientras se desarrolla el sistema, para poder tomar algunas decisiones con cierto grado de urgencia, recuerde que como no se ha seleccionado motor de base de datos aun, es factible trabajar tanto con MySQL o PostgreSQL, por tanto, considerar las respectivas sintaxis de las consultas para el motor que elija trabajar hasta que se defina por parte del equipo. **Al terminar el script que contendrá todas las consultas de este hito, agregarlas a una carpeta para posteriormente comprimirlas junto a todos los códigos fuente**

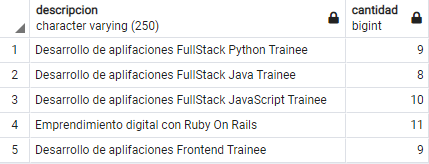
* Se requiere el listado de estudiantes del curso 0012, en donde, se muestre el rut, nombre, apellidos y comuna (nombre de la comuna), la salida debiese ser parecida a la siguiente imagen:



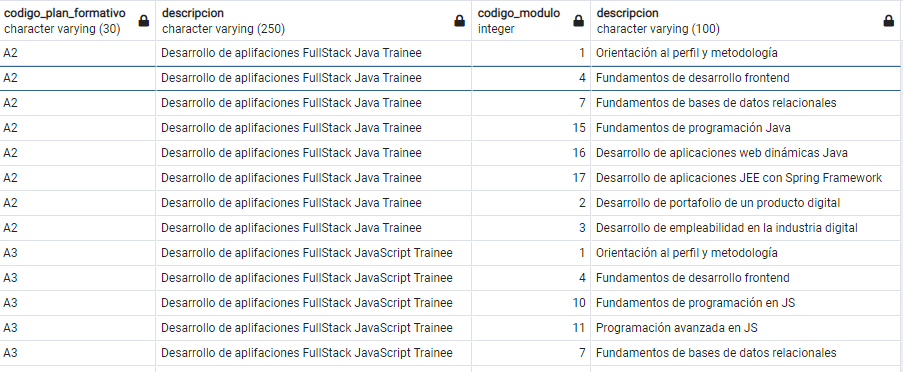
* Se solicita el listado de cursos de todos los cursos existentes hasta el momento, se necesita mostrar el código del curso, fecha de inicio y termino, la descripción del curso (a que plan formativo corresponde, fullstack java, fullstack js, etc) y la duración en horas de cada uno, el reporte se debiese ver similar a lo siguiente:



* Generar un reporte que muestra cada plan formativo con la cantidad de módulos de los que se compone, la salida debe ser similar a la siguiente:



* Se necesita mostrar el listado de módulos de los que se componen todos los planes formativos asociados a FullStack, el reporte debe mostrar el código del plan formativo, la descripción del mismo, código de los módulos y la descripción de los mismos, el reporte debiese ser similar a lo siguiente:



1. Construcción de sistema de Estudiantes

Se requiere construir una página web dinámica que permita el listado y búsqueda de estudiantes, tal como se detalle en la siguiente imagine mock-up.



Se pide:

* En la primera combobox se solicita mostrar el listado de regiones, este listado puede ser estático o dinámico obtenido de la base de datos
* En la segunda combobox, el listado de códigos de los cursos, estos deben consultar a la base de datos para la que la combobox se llene de datos dinámicamente
* Al presionar el botón buscar se debe mostrar el listado de estudiantes que cumpla con los filtros planteados en las dos listas desplegables
* El sistema se debe construir en base a la implementación de una API Rest para poder entregar los recursos necesarios desde la base de datos en formato JSON
* Posterior a ello se debe implementar una SPA que permita generar el consumo de los recursos de la API Rest mediante los endpoints configurados y presentar al usuario la interfaz para que pueda interactuar con el sistema

Para realizar el requerimiento, el Scrum Master menciona lo siguiente:

* Utilizar Bootstrap para enriquecer visualmente la página HTML
* Utilizar NodeJS para poder trabajar con JS fuera del Navegador
* Utilizar NPM para la administración de paquetes en su proyecto
* Utilizar ExpressJS para construir el Backend de la aplicación
* Tener en cuenta que el sitio debe ser responsivo al momento de implementar la vista
* Utilizar las tecnologías necesarias para construir un frontend que pueda consumir los endpoints necesarios para poder atender al requerimiento que se debe presentar al usuario