



UNIVERSIDAD DE COLIMA

**Facultad de Telemática
Ingeniería de Software**

Memoria de Práctica Profesional

BrightCoders Ruby on Rails Web Developer
Brightcoders Consulting S.A. de C.V.

Presenta

Miguel Antonio Rodríguez Perez
20141387, 8° K.

Asesores

D. en C. Carlos Alberto Flores Cortes

M. en C. Sara Sandoval Carrillo

Colima, Col. México, 17 de Junio de 2021.

Índice.

Introducción.	3
Descripción de unidad receptora	4
Marco de referencia	6
Justificación	6
Marco Teórico	6
Marco Legal	7
Descripción de la universidad para las prácticas profesionales	7
Reglamento de la facultad para realizar la estancia profesional	8
Artículos generales del reglamento escolar para las prácticas profesionales	9
Aportación del proyecto en la formación del prestador	12
Desarrollo del Plan de Trabajo	13
Etapa 1: Aprendiendo Ruby	13
Diseño web Kata	13
Game of Life kata	15
Resume kata	16
The Ruby Koans Kata	18
The Bowling game Kata	18
Gilded Rose kata - (La rosa dorada Kata)	19
Minesweeper	19
Etapa 2: Usando el framework on Rails	20
El reloj checador Kata	20
Proyecto denuncia ciudadana	22
Requerimientos	23
Tecnologías	25
Conclusiones	27
BIBLIOGRAFÍA	28

Introducción.

En este documento se redactará la experiencia obtenida durante el transcurso de mis prácticas profesionales, realizadas en el programa de BrightCoders Academy para el curso de Ruby on Rails. Se mencionan las observaciones realizadas previamente a la elección de BrightCoders Academy como unidad receptora.

Se estipulan los reglamentos solicitados tanto de la unidad receptora como de la Universidad de Colima para la presentación de la práctica profesional, lo anterior se encuentra redactado dentro del marco legal.

Se describirán de manera detallada las actividades que se realizaron durante la estancia profesional, teniendo un formato sobre un resumen de los retos enfrentados ya sea de manera individual o como equipo, y cómo se resolvieron dichos retos y/o actividades.

La estancia profesional se dividió en 2 fases, una fase fue completamente para aprender el lenguaje Ruby de manera rápida y divertida. Para lograr llevar a cabo lo anterior. Se hizo una planeación de actividades acordes al nivel que teníamos sobre el lenguaje. La primera fase consistió en Katas, las cuales son retos de programación donde nos ponían requerimientos, instrucciones y fuentes de información para poder realizar la kata. En dichas actividades se trabajan temas como programación orientada a objetos, UX/UI, CD/CI, buenas prácticas de documentación con GIT, pruebas, control de calidad, patrones de diseño y a la par desarrollamos y mejoramos nuestras habilidades blandas.

Durante la segunda fase, se vio más el enfoque al framework on Rails, para el desarrollo web de calidad. En esta fase se trabajó más a detalle las habilidades blandas, pues los 2 proyectos abordados durante este periodo fueron en esa modalidad de trabajo en equipo. Tuvimos algunas dificultades como diferencias de ideas, sin embargo como cualquier equipo pudimos resolver y concluir con un buen trabajo.

Se abordaron los logros obtenidos durante las conclusiones. Importante mencionar que todas las tecnologías utilizadas fueron enseñadas de forma autodidacta, sin embargo siempre contábamos con apoyo de algún mentor si de alguna duda se tratase. Aunque por otro lado, al estar trabajando a la par con otros compañeros, las dudas eran mínimas, pues nos tocó una generación muy capaz y muy hábil.

Descripción de unidad receptora

En esta ocasión. El programa ofrecido por BrightCoders fue plenamente remoto. Debido a los problemas sanitarios de COVID. Es decir, que las prácticas y actividades fueron realizadas en línea desde casa. Sin embargo, es importante mencionar que BrightCoders ofrece esta opción de certificación de manera presencial en las instalaciones de MagmaLabs, la cual se encuentra en Plaza San Miguel, Av. Constitución 2035, Local 11-20, Planta Alta, Santa Gertrudis con código postal 28017 en el estado de Colima municipio de Colima, siendo el encargado de BrightCoders es Carlos Alberto Flores Cortés cuyo puesto en la empresa MagmaLabs es Chief Academic Officer.

BrightCoders Academy como lo dice su página oficial es “una aceleradora de talento que brinda a jóvenes con iniciativa y ganas de aprender programas de certificación diseñados para potenciar sus habilidades, mejorar sus posibilidades de acceso a un empleo de calidad y condiciones de vida” (Bright Coders Academy, 2021). Esta empresa actualmente ofrece 4 programas de certificación, React Native Mobile Developer, Ruby on Rails Web Developer, Shopify Storefront Developer y UI/UX designer.

BrightCoders Academy está fuertemente respaldada por la empresa Magma Labs. Una empresa de desarrollo de software reconocida internacionalmente. Tiene presencia en países extranjeros como lo es Estados Unidos y Canadá. En México tienen sedes en los estados de Aguascalientes, Colima, y León. Es una empresa que maneja principalmente clientes extranjeros y que se ha consolidado bastante (Magma Labs, 2021).

Actualmente, BrightCoders acepta participantes extranjeros, pues el programa es remoto. Hay gente de países de Latinoamérica como lo es Colombia, Venezuela, Argentina y Paraguay, y hasta países de Europa como lo es España; por lo tanto es una empresa en la cual personas de diferentes culturas interactúan entre sí de manera remota para realizar las actividades que en los cursos se implementan.

MagmaLabs funda BrightCoders Academy con el objetivo de brindar capacitación a los recién graduados o jóvenes buscando un trabajo en un campo laboral profesional de desarrollo de software. La capacitación que brinda BrightCoders consta de un curso de certificación, en donde los participantes se les sumerge en proyectos donde pueden aplicar conocimientos necesarios para desarrollarse profesionalmente, proveyendo fuentes de información confiables y ayudas y/o mentorías de Ingenieros profesionales de MagmaLabs. A la vez, busca que el participante se integre en un ambiente de trabajo en equipo. Haciendo la retroalimentación y aprendizaje más sano y sencillo.

Los valores de la empresa MagmaLabs son los siguientes, y fueron recabados de su página oficial:

Visión:

“Ser reconocidos como la mejor agencia de clase mundial de diseño y desarrollo de software en Latinoamérica”.

Misión

“Aplicar nuestro talento y capacidades hasta ayudar a nuestros clientes a siempre alcanzar los mejores resultados de negocio posible, mientras nos mantenemos fieles a nuestros valores principales”.

MagmaLabs es una compañía con principios y valores de desarrollo ágil. Puesto que se enfocan en brindar calidad al cliente, y reinvierten en la capacitación de sus empleados.

Marco de referencia

Justificación

Con el objetivo de cursar una estancia profesional en una empresa que me ayude a crecer profesionalmente dándome bases sólidas para crear software de calidad y permitiendo desarrollar los conocimientos obtenidos en el transcurso de mi carrera universitaria. Decidí unirme al programa de BrightCoders Academy. Este programa es bien reconocido por el alumnado de la Facultad de Telemática de la Universidad de Colima, además de también ser reconocido por la comunidad de desarrolladores de Colima. Cabe mencionar que BrightCoders fue fundado por la empresa MagmaLabs, la cual es reconocida internacionalmente y en ella trabajan personas profesionales y expertos con experiencia en desarrollar proyectos de software centrado en la calidad. Es decir que al cursar su programa se adquiere una retroalimentación de profesionales de MagmaLabs con experiencia en proyectos extranjeros.

El programa que se eligió fue para Ruby On Rails Web Developer (Desarrollador Web de Ruby on Rails). Se optó por este programa debido a la popularidad del framework en el estado de Colima debido a que es utilizado por empresas importantes del estado, como MagmaLabs, Michelada y Tango. Además de otras más a nivel mundial. El framework on Rails ofrece bastante agilidad para crear Proyectos de calidad.

Marco Teórico

Durante las prácticas, no se vio un tema meramente concreto, sino más bien fue un conjunto de temas que se pusieron en práctica al momento de desarrollar los proyectos planteados. Cada proyecto tenía un objetivo y varios temas a desarrollar y/o practicar, la información de los temas a ver venían en los repositorios con enlaces a artículos de blogs o foros importantes o artículos de los mismos ingenieros de MagmaLabs. Tema que no se entendía bien, se solicitaba un taller y un mentor lo abordaba y explicaba.

En los proyectos se pusieron en práctica temas como Mobile First Design, Git flow best practices, Arquitectura MVC, Ruby On Rails Active Storage, Programación Orientada a Objetos con Ruby, gemas, etcétera. También se tocó un poco de Javascript.

Marco Legal

Descripción de la universidad para las prácticas profesionales

La Práctica Profesional es una actividad temporal y obligatoria que puede realizarse en programas universitarios, o en instituciones del sector público o privado y social, que cuenten con un programa aprobado por tu facultad o escuela y registrado en la DGSS y PP.

Son de naturaleza uni, multi o interdisciplinario y pueden ser coordinados por facultades, escuelas o dependencias de la Universidad de Colima.

Requisitos:

- Ser alumno regular inscrito en alguno de los planes de estudio de la Universidad de Colima.
- Haber acreditado el Servicio Social Constitucional y contar con la constancia oficial correspondiente.
- Registrarse en alguna de las unidades receptoras aprobadas por el plantel y la Dirección General de Servicio Social a través de la solicitud y plan de trabajo.
- Cumplir con un mínimo de 400 horas en un periodo no menor a cuatro meses ni mayor a un año.

Documentación:

- Solicitud y plan de trabajo firmados por el alumno, el plantel y la unidad receptora o responsable del programa.
- Carta de presentación expedida, firmada y sellada por el plantel

- Carta de aceptación expedida, firmada y sellada por la unidad receptora (titular o responsable del programa según corresponda).
- Para profesional asociado cubrir 200 horas en un período como mínimo, acumulables para la licenciatura.

Reglamento de la facultad para realizar la estancia profesional

Tienes derecho a:

- Recibir orientación y trato respetuoso por parte del responsable de práctica profesional de tu facultad o escuela y del personal de la institución donde la realices.
- Que se respete el horario, lugar, programa de trabajo y actividades que se establezcan en el programa de práctica profesional registrado y aprobado. Cualquier cambio deberá ser aprobado por tu unidad responsable de práctica profesional.
- Desarrollar la prestación en condiciones de seguridad e higiene, en un espacio físico adecuado, con material, información y equipo para el desarrollo de tus actividades.
- Recibir los apoyos profesionales y académicos estipulados en el programa.
- Recibir oportunamente la documentación requerida para el registro, seguimiento y conclusión de tu práctica profesional.

Tienes la obligación de:

- Inscribirte en los programas de práctica profesional aprobados por tu facultad o escuela y registrados en la DGSS y PP.
- Revisar que la empresa esté registrada en el sistema, si no está registrada entonces enviar el formato de registro de proyecto y documentos de la empresa a mglenda@ucol.mx para que solicite el registro de la empresa a la Dirección Gral. de Prácticas Profesionales.
- Realizar en tiempo y forma los trámites administrativos de práctica profesional en tu facultad o escuela.

- Realizar en tiempo y forma las actividades señaladas en el plan de trabajo y de conducirse de manera profesional y ética.
- Presentar la memoria de las actividades realizadas durante la prestación de tu práctica profesional, cumpliendo con los lineamientos para la realización de la misma.
- En caso de interrumpir tu práctica profesional, deberás notificar por escrito las causas.
- Respetar las disposiciones legales y políticas que establece la U. de Colima y las de la dependencia donde la realices.
- Informar al responsable de práctica profesional de la facultad o escuela correspondiente, si durante tu práctica profesional no se respetan tus derechos, o si se presenta algún problema que afecte la prestación.
- Informar a tu asesor en la empresa las fechas, los requisitos y formatos que deberás entregar a la Facultad para cumplir con los procesos académico y administrativo correspondiente a la Práctica y Estancia Profesional.
- Cumplir con las fechas de entrega señaladas y tomar las consideraciones necesarias para cumplir en tiempo y forma.

Artículos generales del reglamento escolar para las prácticas profesionales

Artículo 1. Los presentes lineamientos tienen como objetivo regular las etapas, pautas, requisitos y criterios que deben observarse en las actividades académicas y los procesos administrativos relacionados con la realización y acreditación de la práctica profesional, en atención a las disposiciones establecidas en la legislación universitaria vigente.

Artículo 2. Los ordenamientos jurídicos institucionales que sustentan los presentes lineamientos son los siguientes:

- Ley Orgánica de la Universidad de Colima: artículo 5, fracciones III y V.
- Reglamento de Educación de la Universidad de Colima, aprobado por el H. Consejo Universitario el 12 de marzo de 2015, mediante el Acuerdo 2 de 2014.
- Reglamento Escolar de la Universidad de Colima, aprobado mediante el Dictamen con proyecto de Acuerdo 1 de 2016, que fusiona los reglamentos

escolares de educación media superior (2006), educación superior (2006) y el de posgrado (2007 y sus modificaciones de 2010). 6 Lineamientos para la realización de la práctica profesional.

- Acuerdo 20 de 2014, por el que se ratifica la creación de la Dirección General de Administración Escolar.
- Acuerdo 28 de 2014, por el que se ratifica la creación de la Dirección General de Servicio Social y Práctica Profesional.
- Acuerdo 40 de 2014, por el que se autoriza la implementación de la mejora regulatoria de la Universidad de Colima y su programa respectivo.

Artículo 3. Los presentes lineamientos son de observancia obligatoria para las dependencias universitarias y planteles de educación superior que intervienen en la realización de la práctica profesional, la cual comprende la planeación, desarrollo, seguimiento y acreditación, así como para los practicantes que las realicen y las dependencias receptoras en términos del registro de proyectos, derechos y obligaciones en su calidad de beneficiarias, incluyendo:

- Planteles de educación superior.
- Los estudiantes de los programas educativos de licenciatura, en lo que corresponda.
- Las dependencias relacionadas con la administración escolar y con las actividades que aquí se determinan, entre ellas:
 - La Coordinación General de Docencia.
 - La Dirección General de Educación Superior.
 - La Dirección General de Administración Escolar y sus direcciones regionales de administración escolar.
- Las dependencias receptoras beneficiarias incluidas en los grupos de:
 - Dependencias de gobierno, en sus ámbitos: federal, estatal o municipal.
 - Organismos no gubernamentales, descentralizados e instituciones de beneficencia pública o asistencia privada.
 - Organismos del sector productivo y privado.
 - Dependencias universitarias.

Artículo 4. El desconocimiento de los Reglamentos de Educación y el Escolar de la Universidad de Colima, así como de los presentes lineamientos, no podrá ser utilizado como argumento válido para evitar el cumplimiento de las disposiciones establecidas en estos documentos.⁷ Lineamientos para la realización de la práctica profesional Vigilancia y evaluación de los lineamientos

Artículo 5. La dependencia responsable de la aplicación y seguimiento de estas disposiciones es la Dirección General de Servicio Social y Práctica Profesional, con la colaboración de las direcciones generales de Educación Superior, Administración Escolar y la de Relaciones Internacionales y Cooperación Académica, así como la supervisión de la Coordinación General de Docencia y la Coordinación General Administrativa y Financiera.

Artículo 6. Las Direcciones Generales señaladas en el artículo anterior, derivado de la evaluación de los Lineamientos para la Realización de la Práctica Profesional, podrán proponer reformas, adiciones o modificaciones, cuando se compruebe la no adecuación de los mismos a la realidad institucional o se presente la necesidad de incorporar nuevas disposiciones.

Artículo 7. Para efectos de los presentes lineamientos, se entiende por:

- Coordinador de práctica profesional: directivo del plantel que tiene bajo su responsabilidad la organización de los procesos de práctica profesional, incluyendo desde su planeación y hasta la acreditación, en un programa educativo determinado.
- Practicante: el estudiante que realiza actividades de prácticas profesionales en una unidad receptora.
- Tutor de práctica profesional: profesor de tiempo completo o asignatura, designado por el director del plantel para atender a un grupo de practicantes, inscritos formalmente en el programa de práctica profesional.
- Unidad receptora: entidad del sector público, privado o social donde el estudiante es aceptado para realizar su práctica profesional.

(Universidad de Colima, 2016).

Aportación del proyecto en la formación del prestador

Cursar mi estancia profesional en el programa de BrightCoders fue una experiencia gratificante e invaluable, se adquirió una filosofía ágil y pude tener retroalimentación de profesionales. Además de que mis compañeros aportaron herramientas y conocimientos que ellos previamente habían adquirido en sus facultades o experiencias anteriores. Logré obtener grandes consejos, información y experiencias de otros camaradas, con las pláticas que teníamos en la oficina virtual de BrightCoders. Aprendí a trabajar en equipo, desarrollé habilidades de comunicación que no tenía anteriormente y logré poner en práctica conocimientos que ya había desarrollado y tuve la oportunidad de afinar durante esta estancia profesional.

Aprendí de fracasos o errores que obtuvimos al trabajar en equipo, rescantando el como podemos evitar estar en la misma situación de nuevo. Logramos aplicar la retroalimentación obtenida de los mentores y así mejorar nuestro nivel de calidad en los proyectos concluidos.

Desarrollo del Plan de Trabajo

Etapa 1: Aprendiendo Ruby

I. Diseño web Kata

En esta actividad se requirió hacer una landing page aplicando el diseño *mobile-first*, es decir que sea responsiva y se adapte empezando por pantallas pequeñas a pantallas grandes como la de un monitor normal o la pantalla de una laptop.

Tecnologías implementadas:

- HTML 5
- JAVASCRIPT
- SASS (pre-procesador de estilo)

repositorio final: <https://bc-mar-21-ror.github.io/kata-01-diseno-web-team001/>

Build The Community Your Fans Will Love

Huddle re-imagines the way we build communities. You have a voice, but so does your audience. Create connections with your users as you engage in genuine discussion.

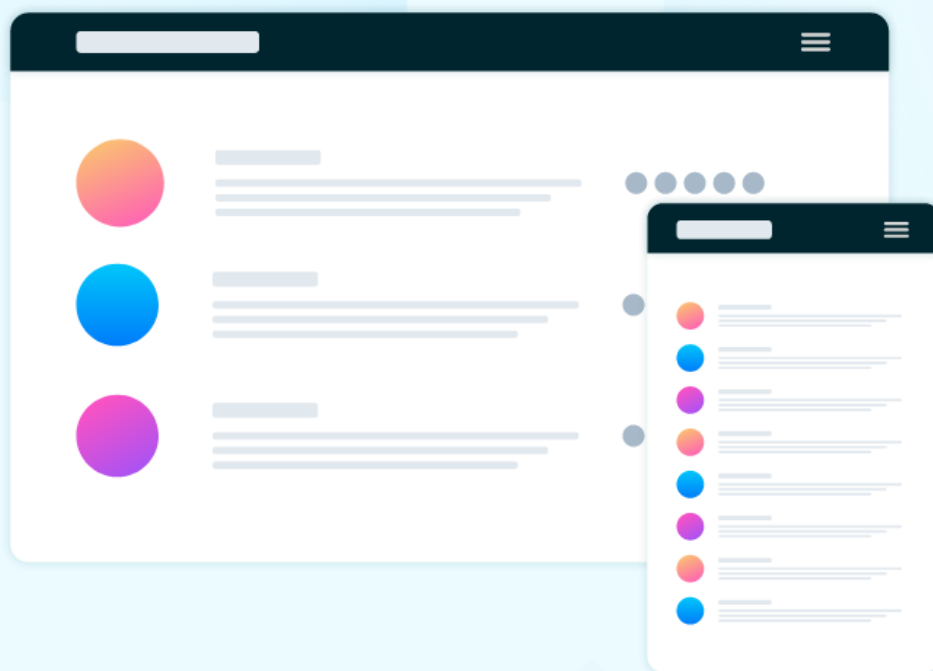
[Get Started For Free](#)

Figura 1. Resultado final, de la practica Diseno web Kata.

Aplique conocimientos adquiridos de la universidad para buscar herramientas que me faciliten el desarrollo de la problemática. Encontré un código que se asemejaba a las funcionalidades de bootstrap 4. Utilice este código y me fue más fácil realizar la actividad con puro html.

II. Game of Life kata

En esta actividad se requieren conocimientos previos del lenguaje de programación ruby. La actividad consistía en crear un algoritmo que simulaba un ambiente en el cual existen células, que, de acuerdo con unas reglas matemáticas, algunas células viven, otras mueren y/u otras se multiplican, todo esto dependiendo de variables como, tamaño del tablero / matriz (x , y), a como inicial de las células vivas y la cantidad inicial de las mismas. Todo esto pueden ser variables aleatorias o ingresadas por el usuario, esto último se realizó para poder documentar pruebas de una manera más sencilla y más limpia, un acomodo inicial y cantidad de células en el ambiente. Las reglas eran las siguientes:

- Una célula con menos de 2 vecinas muere por baja población.
- Una célula con más de 3 vecinas muere por sobrepoblación.
- Una célula con 2 o 3 vecinas pasa viva a la siguiente generación.
- Cualquier célula con 3 vecinas vivas revive.

Lo importante de esta actividad es que se practicó el trabajo en equipo debido a que en un plazo de 5 días, que fueron los días dados para terminar la actividad, los primeros 3 días los equipos cambiarían, de este modo los practicantes estarían día con día en un equipo diferente, compartiendo ideas diferentes y soluciones diferentes para el problema. Así como experiencia y conocimiento previo de cada uno. En los últimos 2 días, nos mantuvimos como equipo, la razón fue que era más sencillo terminar la actividad en 2 días con el mismo equipo. también cada día debía de trabajar nuevamente en la actividad pero debía de hacerse mejor que el día anterior, además de que el equipo debería de apoyarse en el uso de una herramienta para la calidad de código llamada Rubocop, la cual evalúa el uso de las buenas prácticas y convicciones del lenguaje Ruby. También para esta actividad se introdujeron las pruebas unitarias con Rspec, un módulo de ruby que permite realizar

pruebas unitarias las cuales se deberían probar y servir para guiar el desarrollo del programa, siguiendo así la metodología de TDD. Para poder realizar la actividad se nos brindó recursos en internet, los recursos fueron desde documentación de las gemas utilizadas, hasta artículos sobre la metodología TDD, y buenas prácticas de GIT. Se aplicó la programación orientada a objetos y se aplicaron los principios "SOLID". Estos principios ayudan a separar las responsabilidades y hacer más consistente, modulable y limpio la estructura del código del programa. De esta actividad se obtuvo el reforzamiento y aprendizaje de los fundamentos de la lógica de programación y el lenguaje Ruby, además de increíble retroalimentación por parte de los camaradas y fue más fácil el entender el lenguaje al trabajar con otros simultáneamente. Para esto utilizamos liveshare con la extensión de visual studio code además de que se introdujo por primera vez el uso de herramientas para el aseguramiento de la calidad del código buenas prácticas de desarrollo.

Tecnologías:

- Lenguaje de programación Ruby
- Framework para pruebas [RSpec](#) ó [Minitest](#)
- Rubocop, RubyCritic y SandiMeter para calidad de código
- Línea de comando

III. **Resume kata**

En esta breve actividad era hacer un resumen siguiendo un estilo brindado por los mentores del programa y aplicando el modelo dreyfus. Leímos recursos brindados por ellos y aplicando lo aprendido en ellos logramos realizar un currículum utilizando la herramienta [online-cv](#).

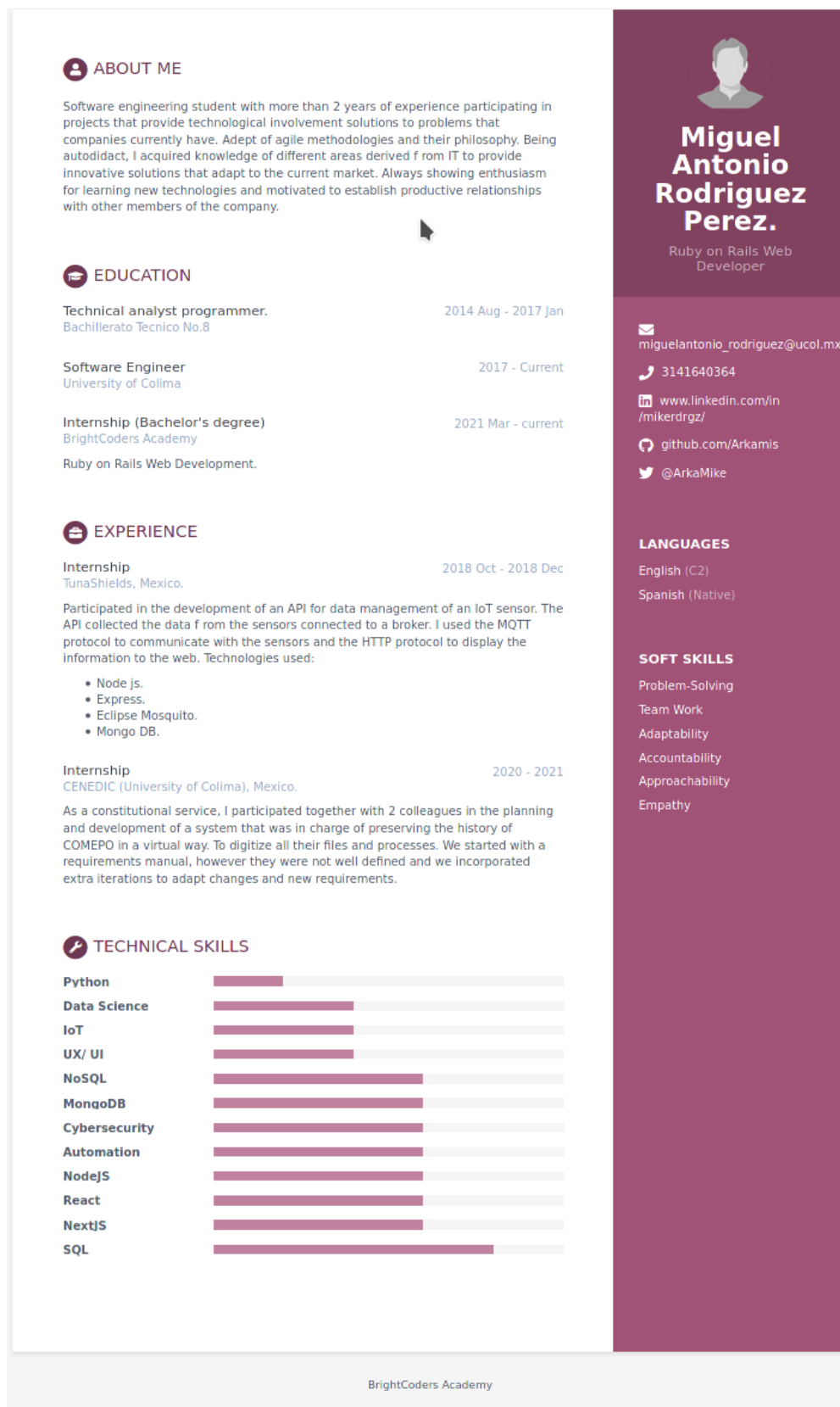


Figura 2. Actividad de Curriculum terminado.

IV. The Ruby Koans Kata

Este kata consistía en un conjunto de pruebas automatizadas, las cuales contenían un conjunto de métodos a ser probados que a su vez enseñaban temas importantes del lenguaje ruby. En cada prueba se debía colocar los valores que harían a la prueba ejecutarse correctamente. De este modo se buscaba que se entendieran los temas de lenguaje de tal modo que se comprendan los resultados correctos de las diferentes funciones, métodos u operaciones en el lenguaje Ruby. Esta actividad se trabajó de manera individual, proporcionando la documentación del lenguaje Ruby y la página oficial de estos retos llamados Koans. Con esta actividad se logró solidificar conocimientos y temas del lenguaje Ruby, sentando mejor las bases en este lenguaje y reforzando temas de la programación en general. Esta actividad fue muy importante para poder entender ruby en general, y es una manera rápida de abarcar todo el lenguaje y conceptos de POO (programación orientada a objetos) en general.

V. The Bowling game Kata

Este kata consistía en el desarrollo de un juego de boliche, se tenía que mostrar el tablero de puntajes respetando las reglas del boliche, estas estaban explicadas en el repositorio de la actividad, el usuario debe poder realizar tiros y el programa debería calcular los puntajes de manera automática. Para esta actividad se asignó a los practicantes en equipos igual que en las otras actividades sin repetir entre equipos. Aprendí otra forma de realizar la actividad y conocí a zsh por un camarada del curso, el cual integre después como consola. También se aplicó la metodología TDD, así como commits atómicos al repositorio.

Tecnologías:

- Lenguaje de programación Ruby
- Framework para pruebas [RSpec](#) ó [Minitest](#)
- Rubocop, RubyCritic y SandiMeter para calidad de código
- Línea de comando.

VI. Gilded Rose kata - (La rosa dorada Kata)

Este kata fue una actividad en equipo, la cual consistía en una tienda que vende diferentes productos los cuales se clasifican en ciertas categorías y tiene un nivel de calidad. Según el tipo de calidad de estos productos y su categoría, es el como su calidad se reduce o aumenta. El código que se proveía era un código mal estructurado e ineficiente, además de que no hacía correctamente lo que debía de hacer. El objetivo era que aplicando la metodología TDD arregláramos el código para aplicar las reglas que afectan a la calidad de los productos. La complejidad de la actividad consistía en refactorizar el código realizado, para cumplir los estándares de calidad de las gemas como rubycritic. Esta actividad ayudó a reforzar los principios **SOLID** y la programación orientada a objetos, además de que motivó aún más el trabajo en equipo obligando a los integrantes el contribuir con ideas para mejorar la calidad del código.

VII. Minesweeper

En este ejercicio se debía crear un código que cuente el número de minas adyacentes a un cuadrado y transforme tableros como este (donde * indica una mina):

```
+-----+
```

```
| * * |
```

```
| * |
```

```
| * |
```

```
| |
```

```
+-----+
```

```
a
```

```
+-----+
```

```
|1*3*1|
```

```
|13*31|
```

```
| 2*2 |
```

```
| 111 |
```

+-----+

Esta kata fue sencilla porque tenía la misma lógica que la del kata del juego de la vida, reutilice el código y pude enfocarme más al kata del reloj checador. La actividad fue individual y se solicitaron los mismos requisitos de calidad que los katas anteriores.

Etapa 2: Usando el framework on Rails

I. El reloj checador Kata

Nos dieron un plazo de aproximadamente 2 semanas para terminarlo. Estuvimos en equipos de 3 sin embargo hubo complicaciones con mi equipo y me volvieron a asignar faltando una semana para entregarlo. Lo interesante de esta kata es que se maneja un calendario con fechas de sprint, siguiendo la metodología de Kanban. Construimos un tablero Kanban en el repositorio de github, está era como un proyecto, creamos issues y las agregamos al proyecto para que salieran como cartas.

Ejemplo del tablero como proyecto:

Cuando realizamos la issue, nos referimos a que implementaremos o corregiremos una funcionalidad. Hacíamos un PR para que otros compañeros brightcoders o mentores lo pudieran checar y validar.

Requerimientos:

Foo Corp necesita un sistema para controlar la asistencia de sus empleados a sus oficinas, necesitan una visibilidad completa de quién fue a trabajar, a qué hora y cuándo se fue. Necesitan una página web que se abrirá en una tableta en el lobby de cada sucursal, donde el empleado pueda ingresar y especificar su número privado y el sistema registrará internamente la hora en que se registró y la hora en que salió.

La aplicación web de asistencia debe cumplir con los siguientes requisitos:

Administración

- Portal de administración protegido con autenticación básica

- Gestionar empleados (operaciones CRUD)

- Desactive empleados en lugar de eliminarlos

- Tienda: correo electrónico, nombre y puesto, número de empleado, número privado

- Gestionar sucursales de la empresa (operaciones CRUD)

- Tienda: nombre, dirección

- Informes

- Asistencia por día

- Tiempo promedio de entrada / salida de los empleados por mes

- Informe de ausencias por mes

Interfaz

- Página de registro de empleados

- Pantalla de entrada de número privado

- Comentarios al empleado sobre el registro exitoso / fallido

Se practicó automatización como CD/CI, esto fue integrado por parte del mentor. Cabe mencionar que tuvimos 3 talleres, uno de GIT, otro de React Hooks y otro de CD/CI con github actions. También durante este periodo, estuvimos revisando PRs de otros camaradas del curso. Aprendimos bastante de solo seguir a camaradas más experimentados con el framework y en seguir buenas prácticas. Compartimos herramientas entre nosotros como la de *git commitzen* para hacer commits con buenas prácticas y persistentes con formato. Lo importante de este kata fue el manejo de la metodología ágil y la documentación del repositorio siguiendo una planificación creada por los mismos participantes del equipo.

Se aprendió a utilizar templates para las issues y los PRs, esto sirvió para ser más expresivos tanto en las funcionalidades agregadas como en la

documentación de los requerimientos de cada actividad.

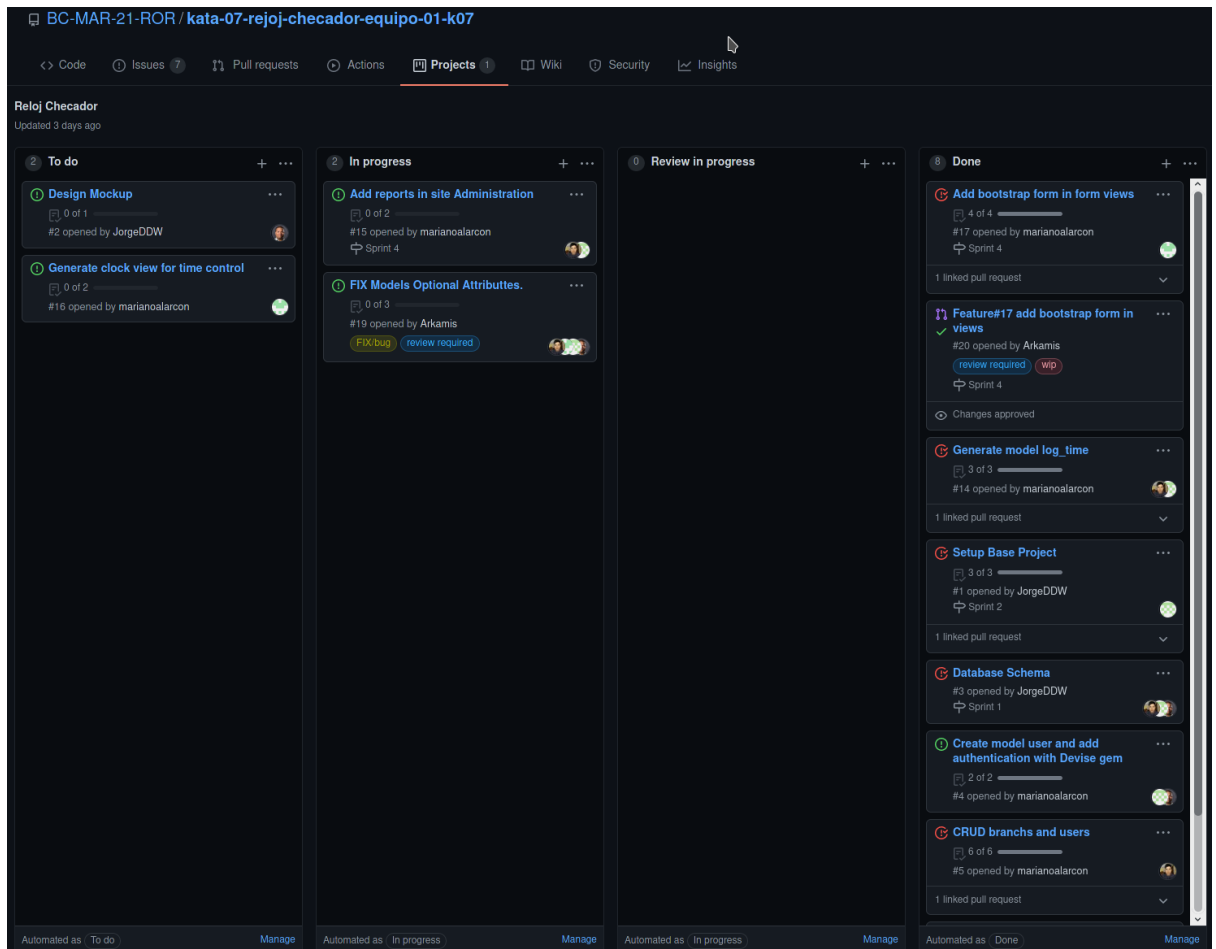


Figura 3. Tablero Kanban en Github projects, mostrando la división de fases y tareas.

Tecnologías usadas:

- Ruby on Rails en el backend
- PostgreSQL como base de datos principal
- CSS: Bootstrap SASS & bootstrap_forms
- Javascript: libertad para utilizar cualquier framework o librería, preferentemente sin jQuery.
- Framework para pruebas [RSpec](#).

II. Proyecto denuncia ciudadana

Este fue el kata más complejo, es una aplicación web con varias funcionalidades básicas de un crud. Trabajamos en modalidad de equipo, siguiendo una metodología

ágil centrada en dar valor importante en las demostraciones de prototipo. Las cuales fueron 5. Cada semana se evaluaba el tablero con las actividades que se planearon, y el estatus de ellas, los estatus eran: to do (por hacer), In progress (en progreso), Review In progress (en PR, pendiente a revisión), Done (realizadas y validadas).

La primera semana nos centramos como equipo en levantar los servicios necesarios para trabajar en equipo con la menor dificultad posible, dando soporte a los requerimientos de integración continua. Para ello utilizamos Heroku y Github Actions. Además de otros servicios como SendGrid para correos, y AWS S3 storage para las imágenes y archivos.

Logramos acabar los requerimientos principales en la semana 3, los bonus en la semana 4 y en la semana 5 nos enfocamos en detalles de usabilidad y experiencia de usuario.

Requerimientos

Una organización civil requiere de una aplicación web mediante la cual se puedan registrar y consultar denuncias ciudadanas. La aplicación web deberá cumplir con las siguientes funciones:

- Registro de usuario
 - Permite a cualquier ciudadano registrarse proporcionando la siguiente información:
 - Usuario. Campo alfanumérico que permite identificar de manera única a cada persona registrada.
 - E-mail. Correo electrónico que sirve de contacto con el usuario.
 - Clave. Campo alfanumérico para la clave de acceso del usuario.
- BONUS POINTS el e-mail debe ser verificado por algún mecanismo, por ejemplo, enviando algún código de confirmación al correo
- Inicio de sesión:
 - Permite a los usuarios registrados y al administrador iniciar sesión. Para iniciar sesión deberán capturar su e-mail y clave.
- Home
 - Inicio. Muestra un listado de las denuncias presentadas en los lugares de interés que el usuario tiene registrados, ordenado cronológicamente de la más reciente a la más antigua.
 - Configuración. Esta opción o pantalla permite al usuario:
 - cambiar su nombre de usuario.
 - cambiar su correo electrónico.

- cambiar su clave de acceso.
- agregar lugares de interés. Los lugares de interés se componen por los campos código postal y asentamiento.
- Mis denuncias. Esta opción muestra un listado de las denuncias que se han presentado ordenadas cronológicamente de la más reciente a la más antigua.
- Agregar denuncia. Esta opción permite agregar una nueva denuncia.
- Inicio
 - El listado de la pantalla de inicio, muestra las denuncias presentadas en los lugares de interés que el usuario tiene registrados, ordenado cronológicamente de la más reciente a la más antigua. La información que el usuario puede ver es:
 - Usuario que hizo la denuncia
 - Área
 - Título
 - Descripción
 - Evidencia(s)
 - Botón que permite otros usuarios manifestar su apoyo a esa denuncia (like)
- Agregar denuncia
 - Permite agregar una nueva denuncia. Para agregar una denuncia el usuario deberá registrarse la siguiente información:
 - Usuario [requerido] El usuario que realiza la denuncia, debe existir la opción de hacer denuncias de manera anónima.
 - Fecha de registro, fecha en que se registra o publica la denuncia.
 - Fecha de hechos fecha en la que ocurrieron los hechos o denuncia.
 - Área [requerido] El área de la denuncia, por ejemplo, seguridad, servicios.

- Infraestructura, etc. Se toman de un catálogo definido en la base de datos.
- Título [requerido] [Un título que identifica a la denuncia.
- Descripción [requerido] Descripción de los hechos o denuncia.
- Ubicación [requerido] Dirección de los hechos o denuncia. Está integrado por lo siguientes campos:
 - Estado[requerido].
 - Municipio [requerido].
 - Código postal [requerido].
 - Asentamiento[requerido].
 - Calle [no requerido].
 - Número [no requerido].
 - La información se toma de un catálogo almacenado en la base de datos el cual puede llenarse con información obtenida de esta página
- Evidencia que se agrega como evidencia. BONUS POINTS Permite agregar más de una imagen o documento PDF.
- Mis denuncias:
 - Permite ver un listado de las denuncias que el usuario ha registrado presentado ordenadas cronológicamente de la más reciente a la más antigua y le permite modificar o eliminar denuncias.

Tecnologías

- Ruby on Rails en el backend.
- PostgreSQL como base de datos principal.
- CSS: libertad para utilizar cualquier framework o librería, preferentemente SASS.
 - Por ejemplo puedes utilizar Bootstrap y junto con bootstrap form para tus formularios.

- Javascript: libertad para utilizar cualquier framework o librería, preferentemente sin jQuery.
- Framework para pruebas Rspec ó Minit
- SendGrid para correos
- AWS S3 Storage como contenedor de multimedia.
- Heroku como Host de Servidor

Conclusiones

La experiencia obtenida de mi estancia profesional en el programa de BrightCoders Academy fue bastante buena, interesante y retadora. Me beneficio bastante poder compartir mis conocimientos adquiridos previamente en mi carrera universitaria. Me permitió desarrollar y aprender habilidades de comunicación y trabajo en equipo que no tenía anteriormente. Me permitió conocer compañeros maravillosos que con sus metas y objetivos, te inspiran a seguir haciendo lo que te gusta.

En cuestión a las actividades y proyectos realizados en este programa, puedo decir que fueron realmente útiles para poder aprender a desarrollar con el framework on rails. Entendí cómo puedo aprender un nuevo lenguaje en poco tiempo y que con la práctica todo es más fácil.

La planificación previa de las actividades del curso me dieron un panorama general de lo que iba a prender, y al terminar la estancia profesional puedo decir que fueron meramente acordes a lo que se estipulo desde un principio. Agradezco la retroalimentación obtenida de los mentores puesto que me sirvió y se que me servirá en un futuro para mi carrera profesional, espero con ansias poder volver a trabajar con los compañeros con los que curse, puesto que la amistad entablada en las prácticas fue completamente sana y agradable.

Cabe destacar que me quedo con la enseñanza del desarrollo ágil del software, haciendo énfasis en que una buena planeación en equipo lleva a reducir tiempo y aumentar tanto la calidad como la productividad. Es importante mencionar que las actividades que realizamos en el programa, nos hicieron desarrollar y aprender nuevas habilidades técnicas. Pero a su vez, también adquirimos habilidades blandas, las habilidades blandas son “el resultado de una combinación de habilidades sociales, de comunicación, de forma de ser, de acercamiento a los demás, entre otras, que hacen a una persona dada a relacionarse y comunicarse de manera efectiva con otros” (Mujica Leiva, 2016). Desarrollar dichas habilidades aporta gran valor a un Ingeniero de software. Algunas habilidades blandas desarrolladas fueron empatía, trabajo en equipo, comunicación, adaptación,

resolución de problemas y creatividad pues adquirí otros puntos de vista para realizar las cosas debido a la diversidad cultural que provee el curso.

Bibliografía

- *BrightCoders Academy*. (2020). Brightcoders. <http://www.brightcoders.com/>
- MagmaLabs. (2020). *MagmaLabs | Ecommerce, Design and Development Experts*. <https://www.magmalabs.io/about-us>
- M.L.J. (2016, 5 diciembre). *¿Qué son las habilidades blandas y cómo se aprenden?* Educrea.
<https://educra.cl/que-son-las-habilidades-blandas-y-como-se-aprenden/#:%7E:text=Las%20habilidades%20blandas%20se%20entienden,de%20manera%20efectiva%20con%20otros%E2%80%9D>.

Asunto: Revisión y Aprobación de la Memoria de Prácticas Profesionales

Por medio de la presente se HACE CONSTAR QUE FUE REVISADA Y APROBADA la Memoria de Práctica Profesional del alumno RODRIGUEZ PEREZ MIGUEL ANTONIO, con número de cuenta 20141387, del Octavo semestre Grupo K de la Ingeniería en Software, en este documento se reportan las actividades realizadas en la Unidad Organizacional: "BrightCoders Consulting S.A. de C.V" colaborando en el proyecto Ruby On Rails Web Developer.

Durante el periodo del 08 de Enero al 18 de Junio del presente año, cubriendo un total de 430 horas, lo que le permite cumplir con los requisitos correspondientes a la materia de Práctica profesional en modalidad de Estancia Profesional.


Chief Academic Officer - Dr. Carlos Alberto Flores Cortés
BrightCoders Consulting S.A de C.V



M. en C. Gerardo Emmanuel Cárdenas Villa
Director de la Facultad de Telemática



M. en C. Sara Sandoval Carrillo
Facultad de Telemática

M en C. Margarita Glenda Mayoral Baldivia
Coordinador de Prácticas Profesionales



Miguel Antonio Rodriguez Perez