



DISEÑO Y CREACIÓN DE UNA PLATAFORMA ACADEMICA ALTERNATIVA

**JULIÁN CAMILO CRUZ SÁNCHEZ
GERSON OBED HERRERA GUTIÉRREZ
MIGUEL LEONARDO URDANETA CAJAMARCA**

INGENIERÍA DE SISTEMAS



ESCUELA DE FORMACIÓN EN INVESTIGACIÓN

BOGOTÁ, 29 DE MARZO DEL 2021

Problema de Investigación

Actualmente, las plataformas enfocadas a la gestión académica, que brindan servicios como inscripción de materias, registro de notas, horario académico, entre otras, cuentan con un problema en su aspecto de usabilidad. Muchas de las plataformas de ese estilo en Colombia presentan cierta falta de empatía con el usuario, sobre todo en los sistemas para el registro de materias o asignaturas y su posterior visualización.

Estas han sido grandes herramientas frente a la educación en una época de cambios y revolución digital, diversifican y cambian el paradigma de la formación en las instituciones al punto de poder ayudar a la presencialidad en situaciones especiales, además, estas plataformas ayudan a la “mejora del trabajo individual, autonomía, organización de trabajos en equipo y la adaptabilidad de los métodos de evaluación.” (Barrera y Guapi, 2018)

¿Qué ocurre cuando estas importantes herramientas tecnológicas no se diseñan bien ni se enfocan en su público objetivo? Además de generar una experiencia de uso complicada, llegan a resultar frustrante al ser usadas, sobre todo en ambientes universitarios como son inscribir materias y visualizarlas. Con todo lo anterior la pregunta a cuestionarse es ¿Cómo diseñar una plataforma amigable e intuitiva que permita suplir las necesidades académicas de los usuarios y sus entidades?

Objetivo general

- Diseñar una plataforma académica que en su vida útil sea amigable e intuitiva con el usuario y que cumpla con las necesidades estudiantiles tanto como con las de la institución.

Objetivos específicos

- Identificar las necesidades estudiantiles e institucionales para la creación de la plataforma que sea eficaz y acertado para la realización de las actividades académicas.
- Organizar la operación de los procedimientos adecuados para la utilización de la plataforma de gestión académica en los diferentes tipos de instituciones estudiantiles.
- Identificar las necesidades actuales de los estudiantes que garanticen un correcto diseño y administración de la plataforma.

Análisis de requerimientos o especificaciones técnicas

Con la construcción de este espacio académico la principal funcionalidad que se busca implementar en el ámbito estudiantil es la de optimizar de manera ostensible el proceso lento y tedioso que experimentan los diferentes *Stakeholders* tanto en la asignación autogestionada de cursos y horarios para los diferentes programas que componen la oferta de las diversas instituciones educativas, como en la modificación, anulación y confirmación por parte de los diferentes departamentos de registro académico involucrados. Se busca modelar a través de la plataforma web y una aplicación móvil de respaldo, un algoritmo sintetizado, plasmando las principales problemáticas recopiladas a través del novedoso estudio guiado por Historias de Usuario (HU) ágiles implementadas mediante el uso de marco de trabajo ágil SCRUM. Dichas historias se ejecutan como descripciones cortas y simplificadas de una característica del espacio a desarrollar, pero desde la perspectiva del usuario enfocada a las funcionalidades que darán solución a los inconvenientes reales del proceso actual. Adicional a lo anterior la plantilla de las historias permite que el principio de simplicidad sea materializado, y se llega a este objetivo a través de 3 parámetros fundamentales: 1. Como: rol de usuario, 2. Quiero: objetivo, 3. Para poder: beneficio.

Partiendo del hecho de que en nuestro mismo entorno institucional, para el estudiante de la Universidad EAN genera cierta molestia la inscripción de materias, pues la plataforma que se usa actualmente, a pesar de que es organizada, al ser usada de forma masiva en los días en los cuales se mantiene la ventana abierta, se torna lenta y da la sensación de que las 3 a 4 horas que habilitan al estudiante no van a ser suficientes, tanto para realizar la respectiva verificación manual de horarios (práctica que es realizada por la mayoría de estudiantes de forma paralela a la inscripción) como para los periodos de espera en el cargue de las funcionalidades en el módulo de inscripción de materias de la página web de la universidad. Por eso resulta importante encontrar una solución estándar que no solo abarque el ambiente académico de la Universidad EAN, sino que va a ser relevante a nivel social para las diferentes universidades a nivel local y nacional, incluso puede ser destinada para su uso en otras instancias académicas como colegios en caso de que este fenómeno de la virtualidad continúe, como es de esperar, logrando así que en todo el entorno académico local y nacional


se ahorre tiempo y espacios valiosos, optimizando procesos y contribuyendo a un ecosistema digital, ágil, sencillo y amigable con el usuario.

Especificaciones técnicas de desarrollo

Las especificaciones técnicas del entorno de desarrollo de la plataforma se conforman del uso de tecnologías como JavaScript como lenguaje de programación principal, estructura y funcionamiento de la herramienta. Para mantener una construcción mediante componentes se hará uso de *frameworks* altamente calificados y revolucionarios en la comunidad del desarrollo web, como *React*, componentes que a su vez están escritos en sintaxis *JSX* y utilizados mediante *SASS* para todo lo relacionado con la hoja de los estilos además del uso de componentes prediseñados de la librería *Ant Design*. El proyecto en su totalidad se irá cargando a un repositorio de GitHub con el fin de realizar control de versiones y mantenimiento del orden a medida que se vayan generando nuevas características y avances en la aplicación. Así mismo, para el servicio de almacenamiento de datos y contenidos multimedia se usará *Firebase*, específicamente sus módulos de *Realtime Database* y *Storage*. De manera temporal se trabajará localmente a través de *local Storage*, pero se proyecta hacer el respectivo escalamiento a consultas en una base de datos alojada en *Firebase* para consultas en tiempo real. Finalmente, para el despliegue de la aplicación móvil se hará uso de tecnología de Aplicaciones Web Progresivas permitiendo así que la ejecución y funcionalidad tanto en navegador como en móvil sea exactamente la misma.




Basados en la filosofía institucional, el marco de investigación del proyecto corresponde al campo “Ciencia, tecnología e innovación”, y la línea corresponde a “Tecnologías de información y telecomunicaciones”, aunque también encierra lo relacionado con línea de “Gestión y diseño de procesos”.

A continuación, se anexan imágenes de la aplicación móvil desplegada en el siguiente servidor público: <https://coursespace.netlify.app/>, desplegada actualmente con la pantalla de inicio y la funcionalidad de registro de usuarios e inicio de sesión.



¡Registrar tus cursos nunca fue tan fácil!

CourseSpace es una aplicación enfocada en la experiencia de usuario y la velocidad en los procesos que este realiza

-  Navegación intuitiva
-  Rápida y eficaz
-  Amigable con el usuario

Los tiempos de carga son casi inexistentes para que puedas navegar y usar la aplicación sin interrupciones.

[Regístrame](#)

Figura 1. Pantalla de inicio. Fuente propia

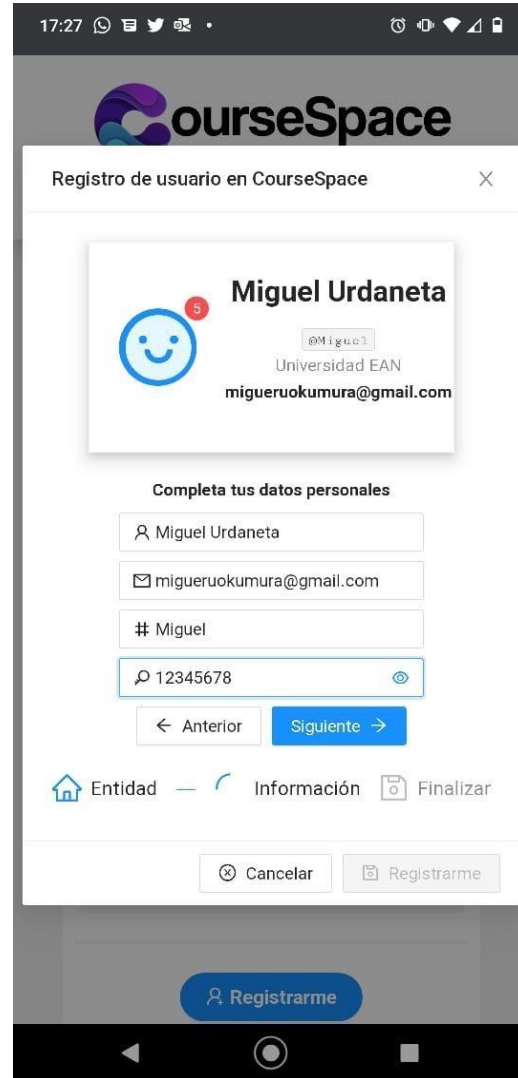
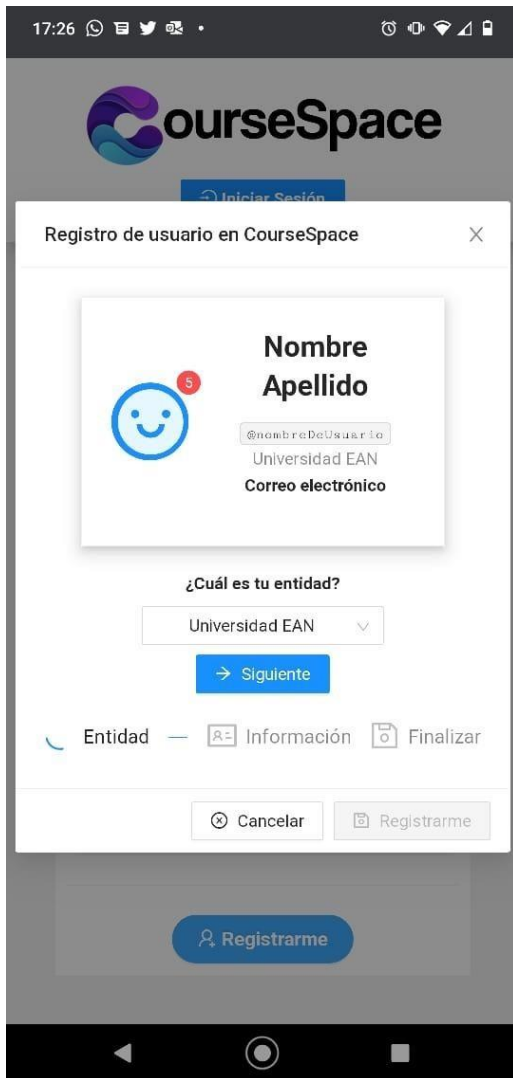


Figura 2 y 3. Registro de usuario. Fuente propia

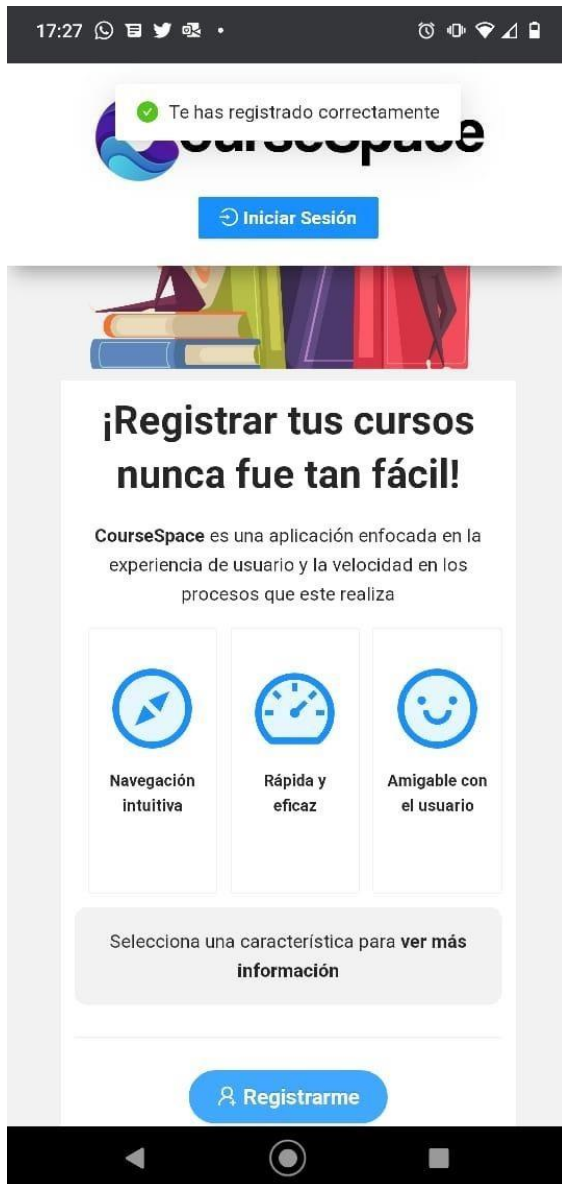
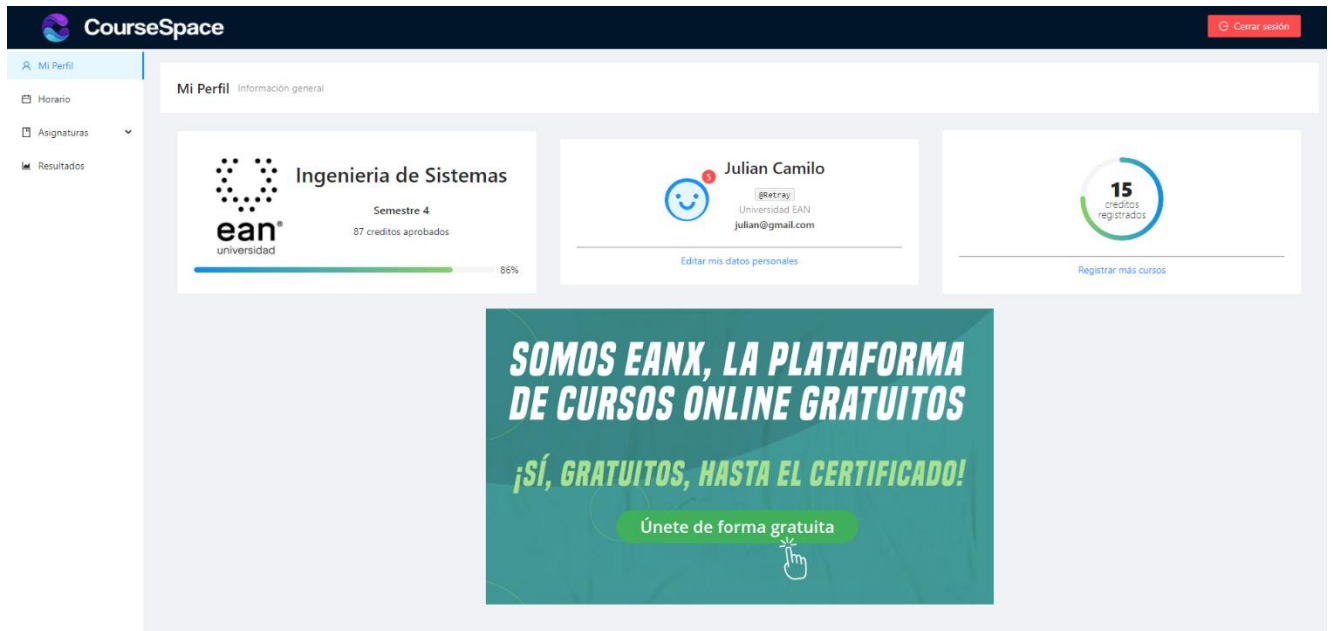


Figura 4 y 5. Inicio de sesión. Fuente propia



CourseSpace Centrar sesión

Mi Perfil Información general

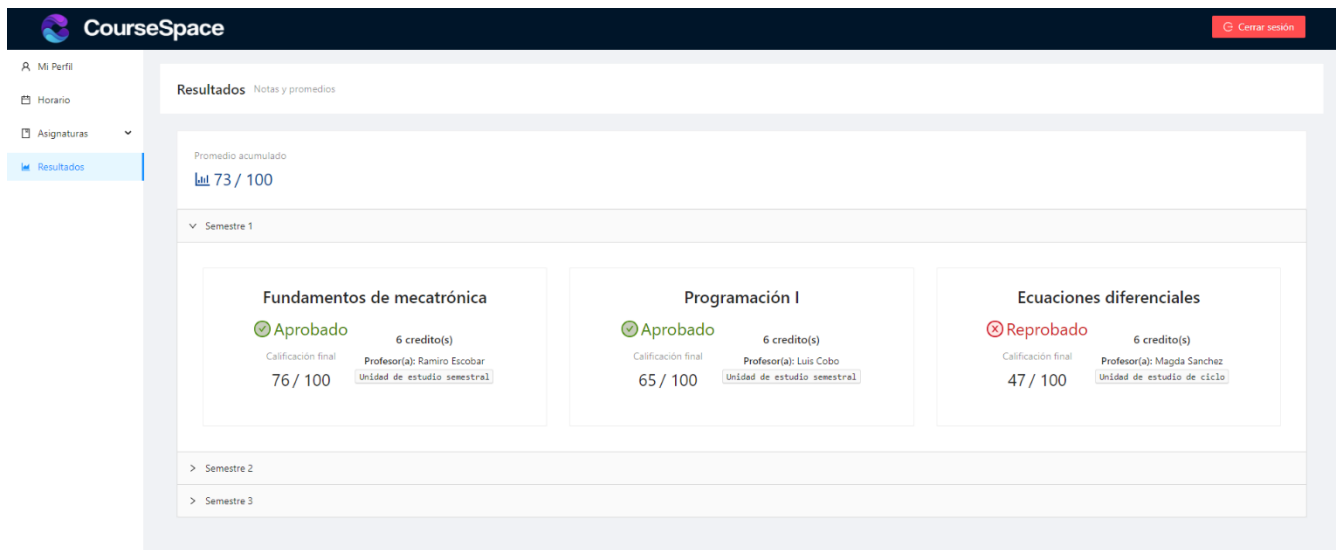
Ingeniería de Sistemas
Semestre 4
87 créditos aprobados
86%

Julian Camilo
@Retray
Universidad EAN
julian@gmail.com
Editar mis datos personales

15 créditos registrados
Registrar más cursos

SOMOS EANX, LA PLATAFORMA DE CURSOS ONLINE GRATUITOS
¡SÍ, GRATUITOS, HASTA EL CERTIFICADO!
Únete de forma gratuita

Figura 6. Dashboard. Fuente propia



CourseSpace Centrar sesión

Resultados Notas y promedios

Promedio acumulado
73 / 100

Semestre 1

Curso	Estado	Calificación final	Profesor(a)	Unidad de estudio
Fundamentos de mecatrónica	Aprobado	76 / 100	Ramiro Escobar	Unidad de estudio semestral
Programación I	Aprobado	65 / 100	Luis Cobo	Unidad de estudio semestral
Ecuaciones diferenciales	Reprobado	47 / 100	Magda Sanchez	Unidad de estudio de ciclo

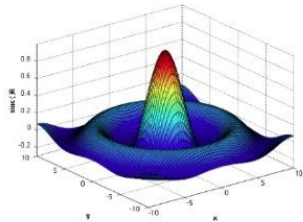
Semestre 2
Semestre 3

Figura 7. Pantalla de resultados. Fuente propia

CourseSpace
Cerrar sesión

- [Mi Perfil](#)
- [Horario](#)
- [Asignaturas](#)
- [Administrar](#)
- [Registrar](#)
- [Resultados](#)

Registrar Nuevos cursos y materias




Cálculo vectorial

Un vector es la representación matemática y gráfica de una magnitud vectorial. Consiste básicamente en una flecha o segmento rectilíneo orientado, es decir, con una determinada longitud, dirección y sentido.

Facultad de ingeniería
6 credito(s)

Juan Pablo Rueda / 6:00 PM-8:00 PM

Registrar




Metodología de la investigación

En su definición pura, la metodología de investigación es el conjunto coherente y racional de procedimientos y técnicas que se aplican de manera ordenada y sistemática en la realización de un estudio.

Cursos transversales
3 credito(s)

Selecciona un grupo

Registrar



Mecánica de fluidos

La mecánica de fluidos es la rama de la física comprendida dentro de la mecánica de medios continuos que estudia el movimiento de los fluidos, así como las fuerzas que lo provocan.

Facultad de ingeniería
6 credito(s)

Selecciona un grupo

Registrar

Figura 8. Registro de nuevo curso. Fuente propia

Marco de referencia

Para el entendimiento de lo que es y representa este proyecto, es necesario interpretar los conceptos que combinándolos componen su realización. En primer lugar, se explicará lo que es el desarrollo de software junto a su ciclo de vida con sus diferentes etapas para luego expandirlo al desarrollo web, el cual se explicará desde el *backend* hasta el *frontend*, explicando acá el diseño UX utilizado como estándar hoy en día. En tercer lugar, está la explicación de la metodología a usarse en toda la duración del proyecto, que en este caso sería Scrum de la cual se explicaran cada uno de sus pasos y de su importancia tanto en la industria como en el proyecto actual. En último lugar, se explicarán los antecedentes que inspiraron o que son muy parecidos al proyecto que se está desarrollando, para así tener una buena base y contextualización de lo que se va a desarrollar.

1. Desarrollo de software

a. ¿Qué es?

Antes de definir el desarrollo de software debemos saber y conocer a profundidad el concepto de software el cual, según Pressman (2010) “El software es el producto que construyen los programadores profesionales y al que después dan mantenimiento durante largo tiempo” (p.1). Incluyendo los programas de computadora, aplicaciones, videojuegos, proyectos de inteligencia artificial, entre otras. Son desarrollados por ingenieros de software con ayuda de un equipo de ingenieros de sistemas, programadores, diseñadores y diferentes personas de diferentes ramas profesionales, en este caso profesionales de la psicología.

Como todo producto tiene un proceso de desarrollo que define cada proceso y paso a paso a realizarse hasta llegar a un entregable para el cliente y su posterior lanzamiento y comercialización todo este proceso se llama desarrollo de software o actualmente conocido como ingeniería de software. Es bien sabido que durante muchos años el software sufrió una

crisis, dada precisamente por la falta de procesos, estándares y regulaciones que otros desarrollos, como el de hardware, ya poseían.

La primera vez que fue usado el término de ingeniería de software fue en una conferencia en Alemania en 1968, desde entonces fueron creándose pequeñas normas y reglas para evitar una nueva crisis, como la primera publicación de la IEEE sobre ingeniería de software en 1972 llamada “Transactions on Software Engineering”. Finalmente, el primer estándar fue planeado desde 1990 y completado en 1995 con la designación de ISO/IEC 12207, teniendo su versión de IEEE en 1996, este estándar tuvo de título “Standar for Software Life Cycle Processes”. Este estándar sigue usándose y actualizándose para definir los ciclos de vida del desarrollo de software, siendo pilar fundamental para la primera versión del libro de conocimiento para la ingeniería de software llamado SWEBOK (Software Engineering Body of Knowledge) en 2004 y sus versiones posteriores. (Fairley D. & Shafer D. 2014, p. XVII).

b. Ciclo de vida

Como ya se mencionó anteriormente el estándar que habla sobre el ciclo de vida de un software es el ISO/IEC 12207 en su versión actual, aquí se define la asignación y cumplimiento de requerimientos por medio de tareas durante varias etapas a seguir hasta que el software deje de usarse.

Teniendo en cuenta lo anterior, Raghu Singh menciona que cada ciclo de vida inicia con una idea o una necesidad a satisfacer por medio del software, esta continua por medio de una arquitectura basada en dos principios: modularidad y responsabilidad. También divide los procesos del ciclo de vida en tres ramas: los procesos primarios, que son obligatorios, estos definen la duración y vida del software siendo estos la adquisición, suministro, desarrollo, operación y mantenimiento; en segundo lugar tenemos los procesos de apoyo, se encargan de, como su nombre lo indica, de apoyar otros procesos en una función especializada,

dentro de estos procesos tenemos la documentación, gestión de la configuración, aseguramiento de la calidad, revisión conjunta, auditoría, verificación, validación y resolución de problemas; por último tenemos los procesos organizativos que se encargan de establecer, controlar y mejorar un proceso de ciclo de vida, estos procesos son la gestión, infraestructura, mejora y formación. (Singh R. S.F. P.1-3).

2. Desarrollo Web

a. ¿Qué es?

El desarrollo web se define por medio de la definición de las aplicaciones web y de la creación de estas mismas para ser visualizadas por medio de un navegador web para obtener completamente los servicios que la aplicación posee. (Jiménez, J. 2013, p. 7)

En los inicios de la internet una web no representaba más que una colección de documentos con limitaciones en lo que a funcionalidades se refiere. Según Mateu, el siguiente paso en la evolución de la web fue el método para confeccionar dinamismo dentro de las páginas, se crearon métodos como el CGI (Common Gateway Interface) para empezar a pasar información entre el servidor y programas externos, pero la ejecución de dichos programas en su época. Fue entonces que se empiezan a desarrollar alternativas, por un lado, se diseñan sistemas más integrados al servidor, por el otro lado se dota al servidor con un intérprete del lenguaje de programación para reducir el tiempo de respuesta. Es aquí donde se expande el número de arquitecturas y lenguajes para este tipo de desarrollo, pero se popularizan sobre todas las que permiten utilizar ambos sistemas, llegando hasta la actualidad del desarrollo web. (Mateu, C. 2004, p.20-21)

Para poder desarrollar las aplicaciones web, es necesario tener en cuenta varias herramientas como un servidor, lenguajes de programación, formatos de diseño, estándares, entre otras, pero todo esto se puede dividir en dos grandes partes: Backend y Frontend.

b. Backend

“El backend es la parte del desarrollo web que se encarga de que toda la lógica de una página web funcione. Se trata del conjunto de acciones que pasan en una web pero que no vemos como, por ejemplo, la comunicación con el servidor.” (Arjonilla, R.2021)

Lo anterior hace referencia a todos los archivos que hacen funcionar un programa o una página web sin que el usuario lo note, es decir todas aquellas funciones que contactan con las bases de datos y los servidores o de aquellas operaciones que se usan para usar cálculos binarios o decimales, define todo lo que debe hacer cada una de las cosas de las aplicaciones web.

El desarrollo del backend se rige a partir de varios diagramas estandarizados denominados como UML (Lenguaje Unificado de Modelado), aquí se define lo que lógicamente la aplicación web debe realizar en cada una de las partes del ciclo de vida del software teniendo en cuenta siempre que UML se basa en el paradigma de la programación orientada a objetos. Durante la creación de todo el UML de un proyecto web es necesario tener en cuenta varios procesos: las vistas, diagramas, símbolos y reglas.

Según Román, “Las vistas muestran diferentes aspectos del sistema modelado. Una vista no es una gráfica, pero sí una abstracción que consiste en un número de diagramas y todos esos diagramas juntos muestran una "fotografía" completa del sistema.” (Román, C. s.f.) Hay diferentes tipos de vistas como la que define la percepción de los actores externos a la funcionalidad del sistema, la que define el diseño lógico de la aplicación, la que muestra los componentes internos de la aplicación, la que muestra la concurrencia del sistema definiendo los errores y comunicación entre

diferentes partes del programa y por último está la vista que muestra la distribución arquitectura física que la aplicación web utilizará.

Después de definir cada una de las vistas utilizadas en el proyecto, definimos los diagramas o gráficas que representan y definen punto a punto cada una de las cosas observadas en las vistas. Así mismo, cada diagrama se compone de diferentes símbolos o elementos que representan conceptos comunes como las clases, las tablas, los usuarios, los casos, entre otras. Y finalizando, se encuentran las reglas que Román las define como las que “Proveen comentarios extras, información o semántica acerca del elemento de modelo; además proveen mecanismos de extensión para adaptar o extender UML a un método o proceso específico, organización o usuario.” (Román, C. s.f.)

c. Frontend

El frontend es todo aquello con lo que el usuario interactúa o todo lo que el usuario puede observar, a esto se le llama interfaz gráfica de usuario (GUI). En desarrollo web las tecnologías comúnmente utilizadas para la creación del frontend son los lenguajes de programación HTML, CSS y JavaScript, pero también hay alternativas como las herramientas de diseño gráfico como lo son Photoshop y Fireworks. El objetivo del desarrollo de esta parte de una aplicación web es entregar una experiencia de usuario satisfactoria. (Valdivia, J. 2016, p.191).

Dentro del desarrollo del frontend existen conceptos que por las empresas y comunidades son estándares y son realmente importantes para toda la experiencia de usuario, de estos destacan el diseño responsivo y el diseño UX.

“Se describe como una técnica de diseño y desarrollo web que, mediante el uso de estructuras e imágenes fluidas, así como de media-queries en la hoja de estilo CSS, consigue adaptar el sitio web al entorno del usuario” (Labrada, E. & Salgado, C. 2013). Lo que definen estas dos autoras es la

forma técnica de lo que significa el diseño responsivo, que traduciéndolo a lenguaje común es la forma de hacer que la interfaz gráfica de usuario se adapte a cada uno de los dispositivos que los usuarios van a utilizar, que la aplicación web se vea bien, organizada y estilizada tanto en una pantalla de computador como en una pantalla de celular o tableta, haciendo que el usuario se sienta cómodo al ver el diseño en general de la aplicación sin importar del dispositivo que utilice. Se debe tener en cuenta siempre un diseño fluido y flexible en la hoja de estilos CSS, parametrizando cada una de las características que componen este archivo, poniendo márgenes, tamaños, entre otras cosas a imágenes, textos, tablas vídeos, básicamente a todo lo que se quiera mostrar dentro de la página o aplicación web. El diseñar responsivamente una aplicación reduce costos, vuelve eficaces las actualizaciones al necesitar una sola plantilla para la producción de la página, mejora la usabilidad y legibilidad de la aplicación y hace muy adaptable la interfaz de la aplicación.

d. Diseño UX/UI

La experiencia de usuario (UX) y la interfaz de usuario (UI) es uno de los temas fundamentales a la hora del desarrollo de aplicaciones web, debido a que estas son la forma y la visión del usuario hacia nuestra página. Estos están correlacionados siempre debido a que un correcto diseño de la interfaz interfiere positivamente en la experiencia que un usuario tiene con nuestra aplicación. (Platzi, 2019).

Para entender un poco sobre UX y como UI es parte de este, hay que ver un poco de la historia de la creación de la experiencia de usuario y como esta es la que mueve el mercado actualmente.

El origen de UX es comúnmente aceptado que se dio a mediados de la segunda revolución industrial alrededor de 1900. Debido al creciente uso de máquinas industriales era importante facilitar el uso y la eficiencia para la mejora de producción, es decir si para un operario era intuitivo el uso de

la máquina, este iba a poder ejecutar sus tareas de manera más rápida, mejorando significativamente su eficacia, naciendo así una de las partes fundamentales del actual diseño UX. El siguiente gran paso para la experiencia de usuario se dio en el año de 1955 con la creación de Disneyland, haciendo que cualquier tipo de experiencia dentro del parque sea totalmente centrada en el usuario, siendo así pionero en desarrollar la experiencia como servicio. Es así como en 1970 llegamos a la creación de la interfaz gráfica de usuario en Xerox Parc, ellos querían aplicar algo denominado diseño centrado en el usuario, una de las partes fundamentales para el desarrollo de la UX, creando y teniendo en el eje central las casillas de verificación, los radio button, los menús y todas las interfaces necesarias para ello. Pero es por la década de los 90s en que se acuñan los términos de “*user experience design*” y “*user experience architect*” por Donald Norman, a su vez, UX empieza a crecer exponencialmente empezando a conocer en el cociente colectivo concepto como usabilidad, diseño de interacción y arquitectura de la información, reforzados en la actualidad con la creación de las aplicaciones móviles y aplicaciones web que usamos día a día. (Álvarez, C. 2016).

Gracias al crecimiento explosivo y exponencial de estos términos ha hecho que se creen diferentes perfiles o profesiones alrededor de estos, tales como UX designer, arquitecto de información, diseñador de servicios, entre otras. A continuación, se explicarán estas profesiones a más detalle para dar una ejemplificación de las diferentes tareas a desarrollarse para un correcto uso del diseño UX/UI

- **UX designer:** este se encarga de la percepción y sensaciones que el uso del producto debe dejar en las personas que lo estén utilizando, este debe asegurar que todos los puntos de vista del usuario sean lo mejor posible hacia nuestro servicio. (Bosada, M. & Ruiz, D. 2018).

- **Arquitecto de información:** “Arquitectura de la información es el arte y la ciencia de estructurar y clasificar sitios web e intranets con el fin de ayudar a los usuarios a encontrar y manejar información” (Rosenfeld, L., Morbille, P. & Arango, J. 2015). Es decir, es una forma creativa, capaz y efectiva de hacer que la información presentada a los usuarios esté bien presentada, organizada y estructurada, que la navegación sea entendible e intuitiva desde el primer uso de la aplicación, producto o servicio. (UXTips, 2018).
- **Diseñador de servicios:** antes que nada, un servicio se puede interpretar como una sucesión de puntos de contacto entre varios individuos, puede ser de forma presencial o por medio, o entre, máquinas que generan una experiencia e intercambio de valor. (Peñaranda, D. 2020).
Entonces según Nielsen Norman group: “el diseño de servicios es la actividad de planear y organizar los recursos de una empresa para mejorar directamente la experiencia de los empleados e indirectamente la de los clientes”. (como se citó en Peñaranda, D. 2020).
Es decir que en el diseño de servicios se debe tener en cuenta de manera estratégica toda la sucesión de contactos con las personas, empleados, maquinaria y procesos para la mejora de experiencia de usuario.

3. Metodología Scrum

En el marco de la metodología Scrum en el cual se basa el desarrollo de la herramienta, se encuentra uno de los procesos de trabajo colaborativo más valiosos y con proyección del mundo de la ingeniería de software, adoptado por un significativo segmento de la comunidad gracias a los resultados orientados a respuestas ágiles obtenidos en los diferentes proyectos ejecutados bajo esta

directriz. El alcance de los proyectos con uso de Scrum tiene un valor agregado debido al poder de decisión, autonomía, responsabilidad y proactividad que otorga a cada uno de los miembros de los diferentes grupos de trabajo, sin necesidad de entorpecer el trabajo en equipo.

Bajo esta metodología los equipos se conforman en promedio por un grupo de 9 personas sin jerarquía definida cuyo grado de responsabilidad es equivalente entre sí y basado en la confianza recíproca (Sutherland, J. y Sutherland, J. J., 2014). De manera que, apelando a los 3 principios nucleares de planeación usados en Scrum, se plantea que el proyecto *CourseSpace* sea desarrollado, lo cual garantiza un cumplimiento mínimo de objetivos además de auto aprendizaje y auto gestión casi al mismo nivel en cada uno de los integrantes del grupo de trabajo. Recordando dichos principios propuestos por A. Davidson y L. Klemme:

- a. *Scrum Meeting* diaria de corta duración en la cual se socializan los avances del día inmediatamente anterior y las del día en curso acompañada de los posibles atenuantes que impidan el cumplimiento de los mismos.
- b. Planeaciones de iteración realizadas en un intervalo de 15 días con el fin de establecer los objetivos a cumplir antes de la siguiente reunión.
- c. Planeación de lanzamiento de versión definida mínimo cada 3 meses con el fin de establecer el alcance y visión del producto. (Davidson y Klemme, 2016).

4. Antecedentes

Existen dos herramientas locales que se toman como referentes de uso universitario y que ofrecen los servicios de auto gestión para las necesidades de registro y control de cursos académicos por parte de los estudiantes. Dichas herramientas evidencian oportunidades de mejora en parámetros de funcionalidad, navegabilidad, interfaz gráfica y déficit ágil de las diferentes ejecuciones en dichas plataformas en los periodos que presentan un alto flujo de navegación por parte del grupo estudiantil e institucional.

- Córdor:

Es la plataforma destinada para servicios estudiantiles de la Universidad Distrital Francisco José de Caldas pero que por percepción de la comunidad estudiantil de dicha universidad se ha identificado un gran número de fallas tecnológicas que afectan el desarrollo correcto de los periodos de alta demanda de uso de la página web, por ejemplo, en época de inscripción de unidades de estudio y por ende la experiencia del estudiante se ve impactada negativamente.

- SAP EAN:

Es la herramienta digital dispuesta por parte de la Universidad EAN para los mismos fines de auto gestión estudiantil. En este caso la plataforma se destaca por una interfaz gráfica mucho más amigable que la herramienta anteriormente descrita, sin embargo, la página SAP presenta las mismas dificultades de insuficiencia para soportar la alta demanda en el uso de la plataforma lo cual no solo genera una mala experiencia al usuario sino que genera una carga operativa en áreas de registro y control pues se intensifican y aumentan los reportes de fallas que deben ser corregidas manualmente o que obligan a reapertura de periodo de inscripciones para cumplir con las expectativas de adición de materias del estudiante.

Análisis de restricciones

1. Ambientales

El proyecto se desarrolla dentro del marco de regulaciones gubernamentales sobre el cuidado ambiental y desarrollo sostenible, al ser un proceso de materialización intangible de ingeniería y entornos digitales, creando afinidad con el principio de contribución a la conservación responsable del medio ambiente, no obstante, la

pandemia por covid-19 ha traído consigo una serie de restricciones locales de protocolos bioseguros, distanciamiento social y autocuidado, por lo que todo el proceso de creación y promoción del producto debe ser netamente virtual y pensado para su funcionamiento en situaciones similares eventuales.

2. Sociales

Una de las principales restricciones sociales que se identifican es la falta de apoyo a emprendimientos surgidos en un marco sencillo y de bajo presupuesto aquí en Colombia. Los proyectos exitosos en su mayoría provienen de alianzas estratégicas previamente formadas. Para respaldar lo anterior es preciso remontarse al informe de la Red de Cámaras de Comercio CONFECAMARAS (2019) *Dinámica de creación de empresas en Colombia*, en el cual se evidencia la creación de 309.463 unidades productivas, de las cuales el 75,7% corresponden a personas naturales y el 24,3% a sociedades. Ahora bien, después de creadas, más de la mitad de disuelven en los primeros cinco años y el 40% lo hace en el primer año siendo en su mayoría las creadas por personas naturales.

3. Capacidad de fabricación

La capacidad de fabricación si bien no depende de factores o elementos externos a la propiedad intelectual de cada uno de los miembros de la asociación en la fase inicial del proyecto, delimita las actividades a la capacidad de desarrollo de software que se pueda entregar en cuanto a horas trabajadas, la cual en promedio puede ser de entre dos y máximo 3 horas diarias para un total de 8 y máximo 10 horas a la semana. Teniendo en cuenta el mantenimiento que requiere la plataforma una vez culminada el tiempo no es tan restrictivo, sin embargo, a la hora de administrar una base de datos para una institución importante y así mismo construir una plataforma robusta y capaz de sostener la capacidad de interacción de esta, el tiempo de completitud del proyecto hasta su fase 100% funcional puede tomar varios meses de labor.

4. Éticas

En cuanto a las restricciones éticas, lo más importante es brindar una solución a nivel general en cuanto a educación se refiere, pero la primera opción de distribución siempre será la institución que está formando al equipo de desarrollo y que potenció las capacidades individuales y colectivas. De hecho, la idea es producto de una falencia en la plataforma de la Universidad EAN identificada con el paso de los semestres. Pero a su vez es deseable que el proyecto trascienda clases y estándares sociales llegando a instituciones que forman personas de recursos limitados y de esta manera contribuir a la construcción educativa del país en aras del desarrollo social y cultural que lleve a la eliminación de la brecha del analfabetismo.

5. Salud y seguridad

Va muy de la mano con las restricciones ambientales en cuanto a la coyuntura que atraviesa la humanidad a causa del covid-19. Es un riesgo latente y para todo tipo de persona el hecho de contraer el virus por lo cual la principal restricción es el cumplimiento de los protocolos de distanciamiento y trabajo conjunto a distancia.

En cuanto a seguridad las restricciones son las establecidas en las recomendaciones por los estándares internacionales como el ISO 1799 y las ISO/IEC 27000, las cuales sintetizan la seguridad de la información como un activo de valor en las empresas en el cual se debe invertir para su protección.

6. Políticas

Son todas las leyes impuestas por los entes reguladores bajo la jurisdicción colombiana como la claridad en las solicitudes de permisos para acceder a dispositivos o guardar y brindar datos a terceros, la correcta utilización y disposición de los recursos a usar sin incurrir en aprovechamiento comercial de dichos recursos a falta de autorización, leyes de protección de imagen y datos de menores, leyes de prohibición sobre estimulación de vida poco saludable o consumo excesivo de

alcohol u otras sustancias, leyes de geolocalización y configuración de privacidad, aceptación de cookies y debida identificación en caso de configurar un modelo negocio complementado por publicidad dentro de la interfaz.

Generación de posibles soluciones

Teniendo en cuenta todas las aplicaciones estudiantiles utilizadas para la gestión de actividades institucionales, como las ya mencionadas Cóndor y SAP, se deben tener varios aspectos para la correcta gestión y administración del producto que se propone, entre estas debe buscarse una solución óptima para la implementación y accesibilidad en diferentes universidades e incluso colegios.

La primer posible solución es generar la aplicación desde cero por medio de herramientas factibles, rápidas y sencillas, como JavaScript y React para frontend y Python para backend. Con un financiamiento por parte de los programas de financiamiento o semilleros de la universidad EAN e implementando servidores para gestión de la aplicación y que cada institución genere servidores para sus bases de datos, o proveer también dichos servidores aumentando el coste que generaría para la institución nuestra aplicación.

La segunda posible solución es adecuar la aplicación para cada una de las instituciones de forma física y desarrollarla con una base preestablecida generando más contacto con las necesidades del cliente, pero retrasando el lanzamiento y hacer que la institución genere los costes que representaría el desarrollo de esta. Se pueden proponer varias tecnologías para su desarrollo, pero también habría que adecuarlas según las necesidades del cliente y las tecnologías y capacidades de hardware que estos estén dispuestos a invertir.

La tercer y última opción es hacer un alquiler de tecnologías para desarrollar dicho tipo de herramientas educativas, tener ya las bases hechas y contratar a terceros para las necesidades que cada institución requiera, pero esto aumentaría todos los costos y generaría una posible tardanza en la solución de solicitudes o errores que se generen por medio de las herramientas de las que se presten el servicio.

Selección de la mejor alternativa

La mejor alternativa es generar la aplicación desde cero por medio de herramientas de fácil uso y baja complejidad como JavaScript y React o Angular para el frontend y lenguajes como Python o Java para el backend, esto debido a que la aplicación web debe poder desarrollarse fácil y rápidamente, además de poder generar una documentación eficaz y al igual que con las pruebas unitarias y estas tecnologías tienen herramientas y bibliotecas que proveen de todo lo anterior mencionado. Si se tiene un financiamiento y apoyo por parte de los semilleros o programas de la universidad como EAN impacta para generar una base sólida respecto a la creación y posterior mantenimiento de servidores, host y base de datos. Además, si las instituciones llegan a proveer la base de datos de los estudiantes, que es lo más común, el coste de mantenimiento físico se reduciría considerablemente y se lograría integrar de forma más rápida toda la gestión y migración de contenido estudiantil, tales como las notas, nombres y asignaturas que vieron o estén viendo los estudiantes sean de universidad o colegio.

Especificaciones de ingeniería para la solución

- JavaScript como lenguaje de programación principal, estructura y funcionamiento de la herramienta.
- *Framework React*,
- Componentes escritos en sintaxis *JSX*
- *SASS* para los estilos
- Librería *Ant Design*.
- Repositorio *GitHub*
- *Firebase* para almacenamiento de datos y contenidos multimedia. (*Realtime Database* y *Storage*).
- *Local Storage* (temporalmente).
- Aplicaciones Web Progresivas
- Diagramación UML

- Metodologías ágiles
- Diseño UX/UI

Dimensionamiento de los componentes

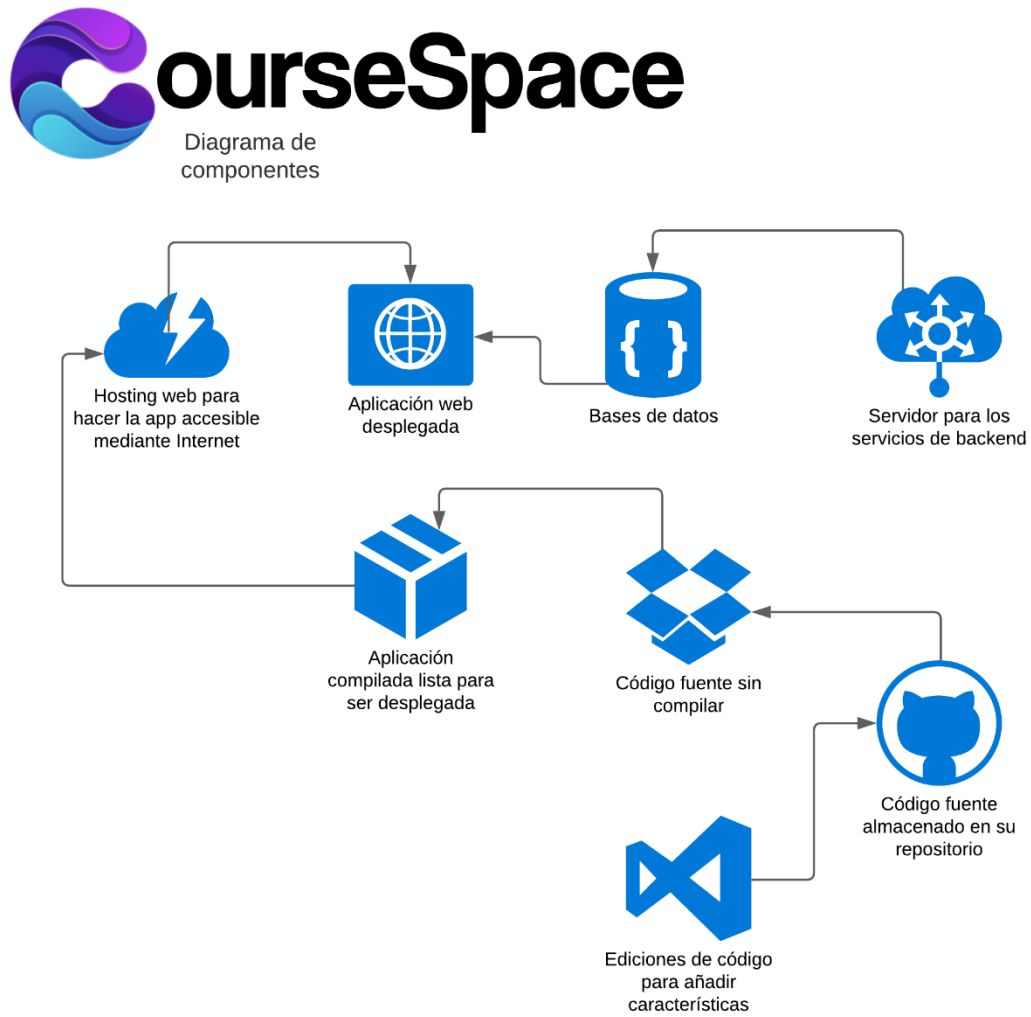


Figura 9. Diagrama de componentes. Fuente propia

Análisis de Costos del Diseño

Para poder definir los costos precisos que este proyecto requiere, se deben tener en cuenta las diferentes características, funciones y requerimientos que lo conforman. Es decir, si se requiere el uso de un lenguaje de programación se debe tener en cuenta el salario de un desarrollador de dicho lenguaje durante el ciclo de vida del software. Para ello se identificarán los diferentes costos y se calcularán por medio de datos y promedios de salarios, hardware y certificaciones que se tienen en Colombia.

Teniendo en cuenta las especificaciones del producto, se debe pronosticar la cantidad de integrantes del equipo de trabajo, cada uno con su especialidad y responsabilidades de ese equipo. Un equipo óptimo para este tipo de trabajo y para las tecnologías a utilizarse, no debe ser muy grande para que las ramificaciones de los repositorios no se compliquen, para tener buena comunicación, no sobrescribir lo que hacen los demás y para poder corregir más fácilmente los errores que se presenten. Ya con todo claro a continuación se presentarán los costos aproximados que podría presentar el desarrollo y vida de la aplicación con los promedios de los costos en el país.

- **Salarios**

- Scrum Máster: el promedio es de 5.048.000 pesos colombianos mensuales. (Talent, 2021).
- Arquitecto de Software: el promedio es de 5.741.398 a 6.747.000 pesos colombianos mensuales. (Computrabajo & Talent, 2021)
- Desarrollador JavaScript/React/NodeJS: el promedio es de 3.181.651 a 4.000.000 pesos colombianos mensuales. (Computrabajo & Talent, 2021).
- Diseñador UX/UI: el promedio es de 1.500.000 a 3.564.722 pesos colombianos mensuales. (Talent 2021).

- **Servicios**

- Repositorio GitHub: 14.938 pesos colombianos mensuales/persona (GitHub, 2021)

- Firebase: 1.090.548,69 pesos colombianos mensuales por 1.000.000 de instalaciones de la aplicación. (Firebase, 2021).

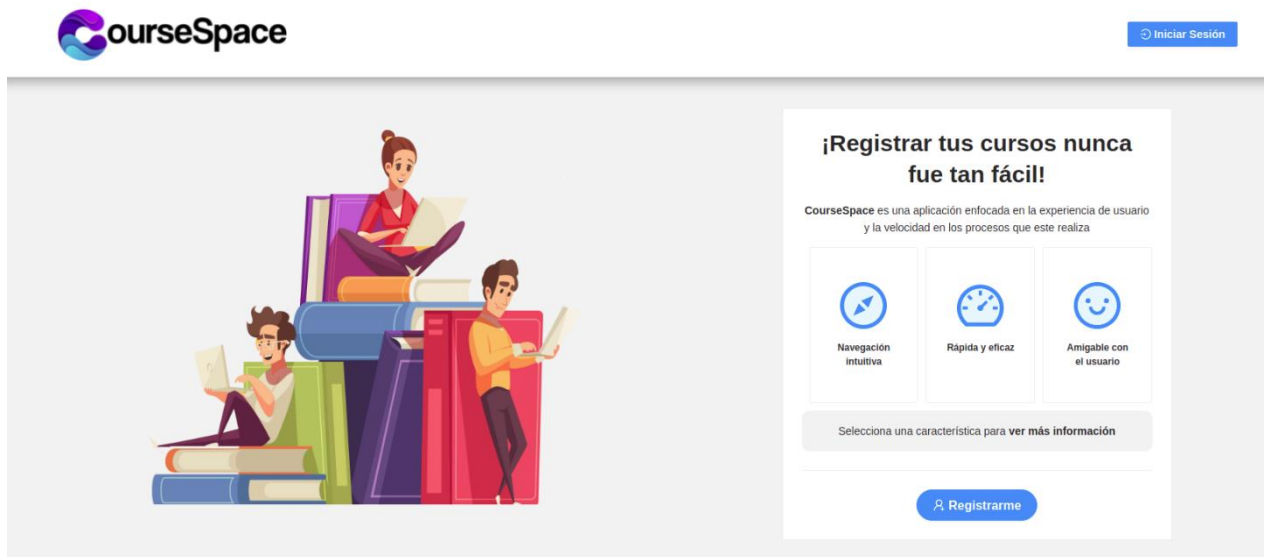
Ahora para definir el costo neto se debe considerar la cantidad de personas dentro del proyecto y que necesiten el uso de repositorio en GitHub y la expectativa de personas que descargarán la aplicación, además teniendo siempre como expectativa de tiempo de seis meses de desarrollo de la aplicación.

Se debe tener en cuenta que los precios a continuación son estimados por los promedios de los datos recopilados, cada uno de estos depende de factores como la experiencia del desarrollador y los salarios que se estén dispuestos a entregar y recibir por los involucrados.

- **Para 5 personas:** teniendo un scrum máster (\$5.048.000), un arquitecto de software (\$5.741.398 a \$6.747.000), un desarrollador frontend JavaScript/React (\$3.181.651 a \$4.000.000), un desarrollador backend JavaScript/NodeJS (\$3.181.651 a \$4.000.000), un diseñador UX/UI (\$1.500.000 a \$3.564.722), tres personas en el repositorio de GitHub (\$44.814), 100.000 descargas de la aplicación (\$109.054,86), dando un total mensual de \$18.806.568,86 a \$23.513.590,86 pesos colombianos y para los seis meses de duración del proyecto un total de \$112.294.138,8 a \$140, 536,270 pesos colombianos.
- **Para 10 personas:** teniendo un scrum máster (\$5.048.000), dos arquitectos de software (\$11.482.796 a \$13.494.000), tres desarrolladores frontend JavaScript/React (\$9.544.953 a \$12.000.000), tres desarrolladores backend JavaScript/NodeJS (\$9.544.953 a \$12.000.000), un diseñador UX/UI (\$1.500.000 a \$3.564.722), ocho personas en el repositorio de GitHub (\$119.504), 100.000 descargas de la aplicación (\$109.054,86), dando un total mensual de \$37.349.260,86 a \$46.335.280,86 pesos colombianos y para los seis meses de duración del proyecto un total de \$223.550.290,8 a \$277.466.410,9 pesos colombianos.

Prototipado

A continuación, se muestran los resultados de la creación del prototipo mediante las herramientas mencionadas anteriormente, aunque no cuenta con una sección de *backend* ni servicios en la nube, cuenta con datos fijos para conocer la funcionalidad, estilo visual y navegación de la aplicación. El repositorio de GitHub se encuentra en: <https://github.com/iRetray/CourseSpace>, y la aplicación se encuentra desplegada mediante Netlify en <https://coursespace.netlify.app/>. Cabe recalcar que la aplicación cuenta con instaladores desde la aplicación para comportarse como aplicación para smartphones.



¡Registrar tus cursos nunca fue tan fácil!

CourseSpace es una aplicación enfocada en la experiencia de usuario y la velocidad en los procesos que este realiza



Navegación intuitiva




Rápida y eficaz




Amigable con el usuario

Diseñada totalmente para que los usuarios puedan **desplazarse de forma autogestionada** y para brindar indicaciones que **orienten mejor** a los usuarios dentro de la aplicación.

 [Regístrame](#)

Registro de usuario en CourseSpace



**Nombre
Apellido**

@nombreDeUsuario
Universidad EAN
Correo electrónico

¿Cuál es tu entidad?


Universidad EAN

→ Siguiente

Entidad — Información — Finalizar

Cancelar Registrarme

Registro de usuario en CourseSpace ✕



Julian Camilo Cruz
@Retray
Universidad EAN
julianmail@gmail.com

Completa tus datos personales

— —

Inicio de Sesión en CourseSpace ×

Ingresar tus datos para ingresar a la aplicación



CourseSpace Cerrar sesión

Mi Perfil Información general

Ingeniería de Sistemas

Semestre 4
87 créditos aprobados

86%

Julian Camilo Cruz Sanchez

@Retray
Universidad EAN
julianemail@gmail.com

Editar mis datos personales

15 créditos registrados

Registrar más cursos

SOMOS EANX, LA PLATAFORMA DE CURSOS ONLINE GRATUITOS

¡SÍ, GRATUITOS, HASTA EL CERTIFICADO!

Únete de forma gratuita

CourseSpace Cerrar sesión

Mi horario Horas y fechas para tus clases

2021 May Month Year

Su	Mo	Tu	We	Th	Fr	Sa
25	26	27	28	29	30	01
02	03	04	05	06	07	08
09	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	01	02	03	04	05

2021-05-06

- Estructuras de datos
- Programación avanzada
- Cálculo II
- Cálculo vectorial

CourseSpace
Cerrar sesión

Mi Perfil

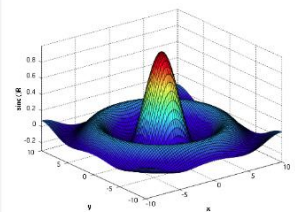
Horario

Asignaturas

Registrar

Resultados

Registrar Nuevos cursos y materias

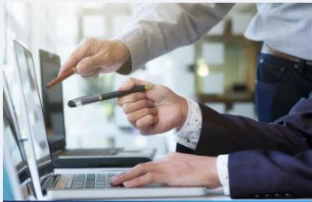


Cálculo vectorial

Un vector es la representación matemática y gráfica de una magnitud vectorial. Consiste básicamente en una flecha o segmento rectilíneo orientado, es decir, con una determinada longitud, dirección y sentido.

Facultad de ingeniería
6 credito(s)

Registrar




Metodología de la investigación

En su definición pura, la metodología de investigación es el conjunto coherente y racional de procedimientos y técnicas que se aplican de manera ordenada y sistemática en la realización de un estudio.

Cursos transversales
3 credito(s)

Registrar



Mecánica de fluidos

La mecánica de fluidos es la rama de la física comprendida dentro de la mecánica de medios continuos que estudia el movimiento de los fluidos, así como las fuerzas que lo provocan.

Facultad de ingeniería
6 credito(s)

Registrar

CourseSpace
Cerrar sesión

Mi Perfil

Horario

Asignaturas

Registrar

Resultados

Resultados Notas y promedios

Promedio acumulado

73 / 100

Semestre 1

Fundamentos de mecatrónica

Aprobado

6 credito(s)

Calificación final **Profesor(a):** Ramiro Escobar

76 / 100 Unidad de estudio semestral

Programación I

Aprobado

6 credito(s)

Calificación final **Profesor(a):** Luis Cobo

65 / 100 Unidad de estudio semestral

Ecuaciones diferenciales

Reprobado

6 credito(s)

Calificación final **Profesor(a):** Magda Sanchez

47 / 100 Unidad de estudio de ciclo

Conclusiones

- La creación de la plataforma en el ejercicio puro de creación y codificación es una labor ardua que requiere de bastante organización en el sentido del *Coworking* por lo que marcos de trabajo de ingeniería de software resultan confortables para cumplir con objetivos y fechas pactadas, incluso afrontando dificultades del ritmo de trabajo que sin guía pueden entorpecer y ralentizar las entregas establecidas.
- A la fecha de la postulación de la solución digital se identifica una seria oportunidad de mejoría en cuanto a la interacción de las diferentes instituciones de educación superior a nivel local y nacional respecto a su comunidad de estudiantes y funcionarios, en cuanto a facilidad de uso, funcionalidad, diseño y disponibilidad se refiere.
- Para desarrollar una aplicación de alcance estudiantil, teniendo en cuenta el enfoque del proyecto y el público objetivo al cual va dirigido el producto final, es importante diseñar un buen esquema de recaudación de información desde el usuario para lograr entender de manera colectiva lo que el público desea de un desarrollo similar. En este caso en particular el uso de UX desde la perspectiva sobre la experiencia de cada usuario permite definir los parámetros iniciales sobre los cuales se cimenta la estructura y diseño más viables y acordes al éxito de la herramienta web.

Recomendaciones

- Se considera importante promover con la misma comunidad estudiantil y apoyo de empresas con experiencia en el sector, el desarrollo de herramientas digitales, sostenibles, funcionales y robustas que permitan acaparar todo el mercado educativo que se ve afectado en periodos de estudio con el fin de acelerar procesos de matrícula e inscripción de cursos y administración de los mismos en las diferentes instancias.
- Se recomienda dar acceso a la plataforma a personal calificado y capacitado para su administración, paralelo a notificaciones institucionales que den nociones básicas del uso de la plataforma a toda la comunidad estudiantil, todo esto una vez se haya completado la construcción del producto mínimo viable.

- Se recomienda constante monitoreo del funcionamiento en los primeros periodos de uso de la aplicación para garantizar evolución de correcto desempeño en el uso y mantenimiento de la misma.
- Se recomienda que los elementos multimedia a cargar en la plataforma se usen evitando formatos de tamaño pesado (en la medida de lo posible) con el fin de optimizar el funcionamiento y tiempo de carga.
- Se puede integrar a futuro en la aplicación todo un sistema de pasarela de recaudo con el fin de ampliar el mercado del producto yendo más allá de instituciones de educación superior, como colegios privados y demás entidades oficiales pertenecientes al sector educativo, dándole otro uso a la plataforma partiendo desde una nueva versión.

Lista de referencias

- Alaimo, M., & Salías, M. (2015). *Proyectos ágiles con Scrum : flexibilidad, aprendizaje, innovación y colaboración en contextos complejos* (2a edición). Kleer.
- Álvarez, C. (2016). Breve historia del UX. Recuperado de: <https://wildwildweb.es/es/blog/breve-historia-del-ux#:~:text=UX%2C%20User%20eXperience%20design%2C%20o,ra%C3%ADces%20bastante%20complejas%20e%20interesantes.&text=Est%C3%A1%20comunmente%20aceptado%20que%20la,a%20mediados%20Segunda%20Revoluci%C3%B3n%20Industrial.>
- Arjonilla, R. (2021) Backend. Recuperado de: <https://rafarjonilla.com/ques-backend/>
- Bosada, M. & Ruiz, D. (2018) UX Designer: qué es, perfil profesional y salidas laborales. Recuperado de: <https://www.educaweb.com/noticia/2018/07/04/ux-designer-es-perfil-profesional-salidas-laborales-18517/>

- Computrabajo. (2021) Salario de Desarrollador JavaScript en Colombia. Recuperado de: <https://www.computrabajo.com.co/salarios/desarrollador-javascript>
- Computrabajo. (2021) Salario de Arquitecto/a de software en Colombia. Recuperado de: <https://www.computrabajo.com.co/salarios/arquitectoa-de-software>
- CONFECÁMARAS. (2019). Crecimiento, supervivencia y desafíos de las empresas de Economía Naranja en Colombia. Recuperado de http://www.confecamaras.org.co/phocadownload/2019/Cuadernos_Analisis_Economicos/CUADERNO%2019%20SEPT%203.pdf
- CONFECÁMARAS. (2019). Dinámica de creación de empresas en Colombia. (p.2) Recuperado de http://www.confecamaras.org.co/phocadownload/2019/Cuadernos_Analisis_Economicos/Din%C3%A1mica%20de%20Creaci%C3%B3n%20de%20Empres%20%20EneDic%202019%2021012020.pdf
- Firebase. (2021). Planes de precios. Recuperado de: <https://firebase.google.com/pricing?hl=es-419>
- Github. (2021). Choose The plan that's right for you. Recuperado de: <https://github.com/pricing>
- IEEE Computer Society (2014). SWEBOK V 3.0 Guide to the Software Engineering Body of Knowledge. Sitio Web: <https://ieeecs-media.computer.org/media/education/swebok/swebok-v3.pdf>
- Jiménez, J. (2013). Aplicaciones Web, Madrid (España). Macmillan Profesional.
- Labrada, E. & Salgado, C. (2013) DISEÑO WEB ADAPTATIVO O RESPONSIVO. Revista Digital Universitaria (14) p. 1 – p.8. Recuperado de:

https://www.ru.tic.unam.mx/bitstream/handle/123456789/2097/art07_2013.pdf?sequence=1&isAllowed=y

- Mateu, C. (2004). Desarrollo de aplicaciones web, Barcelona. Fundació per a la Universitat Oberta de Catalunya.
- Mezinsky, A., López, G., & Palacio, J. (2018). *Historias de Usuario Ingeniería de requisitos ágil* [E-book] (2nd ed., p. 9). Recuperado el 28 febrero de 2021, de https://scrummanager.net/files/historias_usuario_scrum_manager.pdf.
- Platzi (13 de marzo de 2019). UX y UI explicado | ¿Qué hacen los expertos en usabilidad? | Platzi Cursos [Archivo de Vídeo]. YouTube. <https://www.youtube.com/watch?v=zUZSSkRQXQQ>
- Pressman, R. (2010). Ingeniería del Software Un enfoque práctico. Séptima edición. México, D.F: Editorial McGraw Hill.
- Róman Zamitiz, C. (s.f.). TEMAS ESPECIALES DE COMPUTACIÓN. Recuperado de: <http://profesores.fi-b.unam.mx/carlos/aydoo/toc.html>
- Rosenfeld, L., Morbille, P. & Arango, J. (2015). Information Architecture. Cuarta Edición. O'Reilly Media, Inc.
- Singh R. (S.F). International Standar ISO/IEC 12207 Software Life Cycle Processes. Washington, DC. Federal Aviation Administration.
- Suarez Silvestre, C., & Urrego Acosta, C. (2016). *CONSTRUCCIÓN UN PROTOTIPO DE HERRAMIENTA WEB PARA LA GESTIÓN DE LOS SERVICIOS OFRECIDOS EN LA NUBE PRIVADA DEL PROYECTO CURRICULAR DE INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD DISTRITAL* [Ebook] (pp. 82,83). Recuperado el 29 de marzo de 2021, de <https://cutt.ly/dx8Q58g>

- Talent. (2021) Salario medio para Arquitecto Software en Colombia 2021. Recuperado de: <https://co.talent.com/salary?job=Arquitecto+Software>
- Talent. (2021) Salario medio para desarrollador React en Colombia 2021. Recuperado de:
<https://co.talent.com/salary?job=Desarrollador+React#:~:text=El%20salario%20Desarrollador%20React%20promedio,a%C3%B1o%20o%20%2419.692%20por%20hora>.
- Talent. (2021) Salario medio para Diseñador UI en Colombia 2021. Recuperado de: <https://co.talent.com/salary?job=Dise%C3%B1ador+Ui>
- Talent. (2021) Salario medio para Diseñador UX en Colombia 2021. Recuperado de: <https://co.talent.com/salary?job=Dise%C3%B1ador+U+X#:~:text=El%20salario%20Dise%C3%B1ador%20U%20X%20promedio,m%C3%A1s%20experimentados%20perciben%20hasta%20%2429.845>.
- Talent. (2021) Salario medio para Scrum Master en Colombia 2021. Recuperado de: <https://co.talent.com/salary?job=Scrum+Master#:~:text=El%20salario%20Scrum%20Master%20promedio,a%C3%B1o%20o%20%2429.378%20por%20hora>.
- UXTips (18 de abril de 2018). ¿Qué es Arquitectura de la Información? [Archivo de vídeo]. YouTube. <https://www.youtube.com/watch?v=twVvK1eYmig>
- Valvidia, J. (2016). Modelo de procesos para el desarrollo del front-end de aplicaciones web. Recuperado de: <https://doi.org/10.26439/interfases2016.n009.1245>
- Víctor Fernando Barrera Rea y Ana Guapi Mullo (2018): “La importancia del uso de las plataformas virtuales en la educación superior”, Revista Atlante: Cuadernos de Educación y Desarrollo (julio 2018). En línea: <https://www.eumed.net/rev/atlante/2018/07/plataformas-virtuales-educacion.html/hdl.handle.net/20.500.11763/atlante1807plataformas-virtuales-educacion>