***[Jefe de Grupo]*** Vicente Diaz Quezada

Cristobal Olave Herrera

Aldana Bravo Saldivia

Ernesto Nicolas Lagos

Universidad de andres bello

Campus de Antonio Varas

Proyecto: Aplicación de Gestion de Datos “ As You Wish

Informe Avance – 9° Semana

19 de Noviembre de 2020

*Contenidos Generales:*

[1 Introducción: 5](#_Toc56717170)

[2 Descripción Proyecto: 5](#_Toc56717171)

[3 Mapa de Historias: 6](#_Toc56717172)

[4 Gestión de Requerimientos: 7](#_Toc56717173)

[4.1 Alcance 7](#_Toc56717174)

[4.2 Calidad 8](#_Toc56717175)

[4.3 Costo 8](#_Toc56717176)

[4.4 Tiempo 8](#_Toc56717177)

[4.5 Req.Especiales 9](#_Toc56717178)

[4.5.1 CLASE – “Alumno”: 9](#_Toc56717179)

[4.5.2 CLASE – “Clase”: 10](#_Toc56717180)

[4.5.3 CLASE – “Local”: 10](#_Toc56717181)

[4.5.4 CLASE – “Global”: 10](#_Toc56717182)

[4.5.5 Niveles de Permiso: 10](#_Toc56717183)

[5 [ACTUALIZADO] Gestión de Construcción: 11](#_Toc56717184)

[5.1 Diseño 11](#_Toc56717185)

[5.1.1 Estético: 11](#_Toc56717186)

[5.1.2 Uso: 11](#_Toc56717187)

[5.1.3 Estructura: 11](#_Toc56717188)

[5.1.4 Lenguaje Programación: 13](#_Toc56717189)

[5.2 Implementación e Integración: 13](#_Toc56717190)

[5.2.1 Implementación: 13](#_Toc56717191)

[5.2.2 Integración: 14](#_Toc56717192)

[5.3 [ACTUALIZADO] Pruebas de Verificación: 15](#_Toc56717193)

[5.3.1 Logros: 15](#_Toc56717194)

[5.3.2 Fallos de dichos Logros: 15](#_Toc56717195)

[5.3.3 Fallos: 15](#_Toc56717196)

[5.3.4 Plan de Testing: 16](#_Toc56717197)

[5.4 Pruebas de Validación: 17](#_Toc56717198)

[6 Gestión Deployment: 18](#_Toc56717199)

[7 [ACTUALIZADO] Gestión Proyectos: 19](#_Toc56717200)

[7.1 Planificación: 19](#_Toc56717201)

[7.2 Diseño de Estructura principal 19](#_Toc56717202)

[7.3 Diseño de Estructura de Código 20](#_Toc56717203)

[7.4 [ACTUALIZADO] Actividades y sus Responsables 20](#_Toc56717204)

[7.4.1 Establecer funciones finales de Ingreso 20](#_Toc56717205)

[7.4.2 Establecer funcionaes finales de Usuarios 20](#_Toc56717206)

[7.4.3 Establecer .Css 20](#_Toc56717207)

[7.4.4 Establecer .Css 20](#_Toc56717208)

[7.4.5 Monitoreo & Control: 21](#_Toc56717209)

[7.5 Entregables 25](#_Toc56717210)

[7.5.1 Cristobal Olave 25](#_Toc56717211)

[7.5.2 Aldana Bravo 25](#_Toc56717212)

[7.5.3 Nicolas Lagos 26](#_Toc56717213)

[7.5.4 Vicente Díaz 27](#_Toc56717214)

*Contenido de Imágenes:*

[3.1 Imagen – Mapa de Historia 6](#_Toc56717215)

[4.4.1 Imagen - GANTT 8](#_Toc56717216)

[5.1.1.1 Imagen - Estético 11](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%209°Semana.docx#_Toc56717217)

[5.1.3.1 Imagen - Estructura 11](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%209°Semana.docx#_Toc56717218)

[5.1.1.1 Imagen - Estético 12](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%209°Semana.docx#_Toc56717219)

[5.3.1 Imagen – Entregable Simple 15](#_Toc56717220)

[6.1 Imagen - Setting 18](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%209°Semana.docx#_Toc56717221)

[7.2.1 Imagen – Estructura General 19](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%209°Semana.docx#_Toc56717222)

[7.5.2.1. Imagen – Codigo Registrar\_Asistencia 25](#_Toc56717223)

[7.5.2.2 Imagen – Registro HTML 25](#_Toc56717224)

[7.5.3.1 Imagen – Diseño .Css 26](#_Toc56717225)

[7.5.4.1 Imagen – Nuevo Form Pago 27](#_Toc56717226)

[7.5.4.2 Imagen – Pagos Local 27](#_Toc56717227)

[7.5.4.3 Imagen – Pago Estado Mensual 28](#_Toc56717228)

[7.5.4.4 Imagen – Pago Eliminado 28](#_Toc56717229)

*Contenido de Tablas:*

[3.1 Tabla – Mapa de Historia 7](#_Toc56717230)

[5.3.1.1 Tabla – Pruebas Registradas 17](#_Toc56717231)

[7.5.5.1 Tabla – Inicio 21](#_Toc56717232)

[7.5.5.1 Tabla – Alumno 22](#_Toc56717233)

[7.5.5.2 Tabla - Profesor 22](#_Toc56717234)

[7.5.5.3 Tabla – Usuario 23](#_Toc56717235)

[7.5.5.4 Tabla - Clase 23](#_Toc56717236)

[7.5.5.5 Tabla - Local 24](#_Toc56717237)

[7.5.5.6 Tabla - Global 24](#_Toc56717238)

[7.5.5.7 Tabla – No Clasificadas 24](#_Toc56717239)

1. Introducción:

El Proyecto “As You Wish”, es un **Sistema de gestión de diferentes tipos de Datos** creada con el objetivo de Simplificar y ayudar en la recopilación y administración de diferentes tipos de Datos que guardan las Academias de Baile “As You Wish” (Bellavista, La Florida, Providencia, “Casa Matriz” a pasos de Universidad Católica).

A petición del encargado de una de las Academias, y de un Tercero en contacto, en este Documento se detallara de forma precisa y simple todos los Avances tanto logrados como los que se buscan lograr.

1. Descripción Proyecto:

Como se mencionó anteriormente, la Aplicación “As You Wish”, es un Sistema de Gestión de Datos de la Academia cuyo objetivo principal es la Administración de dichos Datos para un mejor y más sencillo modo a la hora de guardar los Datos. Para ello, además de implementar un registro de Datos optimo, es necesario también implementar variadas funciones que den cualquier tipo de Apoyo a su Objetivo principal.

Es por ello, que se busca implementar las siguientes Funciones Listadas:

* Crea, Modificar, Eliminar & Ver Objetos de distintas Clases
* Crear, Modificar, Eliminar & Ver Usuarios que tienen acceso a la Aplicación “As You Wish”, incluyendo datos relevantes para una mejor administración
* Crea, Modificar, Eliminar & Ver Listas de variados Objetos, esto incluye el Clasificarlas, Archivarlas y Administrarlas.
* Introducir o Eliminar participantes de diversas Listas
* Registrar Asistencia de los Alumnos registrados de una Clase de Baile en Especifico
* Clasificar o dividir el uso de la Base de Datos en función de la ubicación de las distintas Academias de Bailes “As You Wish” distribuidas
* Implementar un medio de seguridad que mantenga alejados a los que no están registrados, así como el otorgar un manual de instrucciones para los que si están registrados

1. Mapa de Historias:

Basado en la siguiente Nota de Usuario:

***“Como integrante de la Academia, en calidad de administrar, quiero ser capaz de gestionar una o varias Listas, con el fin de tener todo organizado de manera eficiente y óptima”***

Se decidió crear el siguiente Mapa de Historia con la finalidad de levantar los requerimiento necesario para lograr con el Proyecto “As You Wish”

Imagen que contiene Tabla

Descripción generada automáticamente

3.1 Imagen – Mapa de Historia

|  |  |
| --- | --- |
| Historia | Descripción |
| **Ingresar al Sistema** | Paso primario para el Sistema. El Usuario debe ingresar con Nombre de Usuario & Contraseña luego de haber seleccionado su Residencia para reducir más los Datos. |
| **Seleccionar Opción** | Dispondrá de un gran número de Opciones, los cuales cada uno tiene un propósito único. |
| **Seleccionar Objeto** | Si una de las Opciones involucra Objetos, se tendrá diversas opciones para gestionar o buscar el Objeto. |
| **Configuración de Objeto** | Para modificar el Objeto, este debe cumplir con los Parámetros correctos |
| **Actualizar DATOS** | Una vez registrados DATOS nuevos, el Sistema automáticamente debe modificar todo DATO que guarde alguna relación con el Objeto modificado. |
| **Suspender Sistema** | El Usuario tendrá la Opción de Cerrar Sesión cuando ya no requiera del Programa |

3.1 Tabla – Mapa de Historia

1. Gestión de Requerimientos:
   1. Alcance

**Lo que va ser Incluido:** La Aplicación principalmente debe ser capaz de Controlar de manera eficiente los procesos de registros y administración (Ordenado por Prioridad):

1. Integrar, Modificar, y eliminar la información de diversos Objetos que guardan al tipo de relación con las Academias impartidas de todas las sedes existentes de la academia.
2. Ser siempre capaz de mostrarla de forma ordenada toda información que se desea pedir (Siempre y cuando cumpla con los Requisitos)
3. Manejo simple pero eficiente de las Listas creadas a partir de la información ingresada
4. Manejo de los diferentes tipos de Datos, con respecto a su Área, tanto de información como de Local
5. Manejo óptimo y eficaz para todos los Usuarios registrados
6. Creación de Manual de Usuario

***Se requiere:***

* Conocimiento de Lenguaje de Programación DJAGNO
* Conocer de antemano los Atributos & Objetos que requiere el Cliente
* Cumplir con las Fechas estimadas de Entrega
* Mantener contacto constante con nuestro Contacto de la Academia

**Lo que NO va ser incluido:** La Aplicación únicamente será diseñada para administrar información, por lo que este NO tendrá incluido lo siguiente:

* Capacidad para lograr cualquier medio de comunicación
* Capacidad de manejar cualquier otro Dato que no esté dentro de su Área
* Otorgar permiso de Usuarios a los que no están registrados
  1. ***Calidad***

Actualmente la Aplicación “As You Wish”, se encuentra en su **Versión v1.1** por lo que este ya cuenta con las Páginas Principales para ver todos los Objetos que se registran en la Base de Datos, sin embargo se está aun desarrollando el seleccionar Objetos específicos para Listas o funciones dados unos atributos

* 1. ***Costo***

Al ser **Versión v1.0** este fue desarrollado en un Programa de Lenguaje de Programación gratuito, por lo que por el momento, el **Costo es de $0**, si es que no se está tomando en cuenta los gastos para hacer uso del Internet.

* 1. ***Tiempo***

Creado a partir del 1° Semestre del 2020, este proyecto actualmente se encuentra en desarrollo para lograr una actualización de la **Versión v1.3**, por el momento, la fecha más estimada para lograr un avance notable seria el **23 de Nov.2020**.

Cada Semana, se entregara un Informe detallando los Avances más relevantes del Proyecto excepto cuando en la Semana se requiera de algún Presentación de HITO del Proyecto.

Imagen que contiene Escala de tiempo

Descripción generada automáticamenteA continuación, se presentara un Diagrama de GANTT detallando la planificación y programación de Tareas a lo largo del periodo entre ***2020-04-09*** hasta ***2020-11-23***:

4.4.1 Imagen - GANTT

* 1. Req.Especiales
* Dado que el diseño de la Estructura Base, ya se definió, a partir de este momento ahora se estará desarrollando líneas de Código que se encuentran en las Aplicaciones Django, por lo que es muy poco probable el desarrollo de nuevas Aplicaciones Django.
* Se dio información de que elementos deben ir incluidos para los objetos de la Clase “Alumno”, así como del desarrollo de nuevas Clases. Estos atributos & Clases son tanto de entrada como de salida para visualizarse:
  + 1. CLASE – “Alumno”:

**Input:** (Es la información individual)

* Nombre
* Rut
* Fecha de nacimiento
* Profesión
* Dirección personal y laboral (al menos comuna en la que vive y comuna en que trabaja)
* E-mail
* N° Telefono

Cada uno no puede quedar con información *null*, por lo que se deben programar para que no acepten formas que no cumplen con los requerimientos.

**Output:** (Es la información sobre otros) [SEGUIMOS EN ALUMNO]

* Resultado respecto a compras
* Asistencias [De forma mensual]
* Clases a las que asiste [Incluyendo frecuencia programable por tramos de tiempo (ej. providencia, abril, 5 salsa intermedio martes y jueves, 3 zumba martes]
* Locales a los que asiste [Nombrándolos]

Para las compras estos deben especificar su Tipo [Pase Mensual, Clase Única, etc], incluyendo su comportamiento de compra.

*Asistencia Vs. Compra:* Se nos pide crear una comparativa de cuantas clases vino, con respecto a las que pago. Ej: El Pago por 5 Clases, pero únicamente asistió a 2.

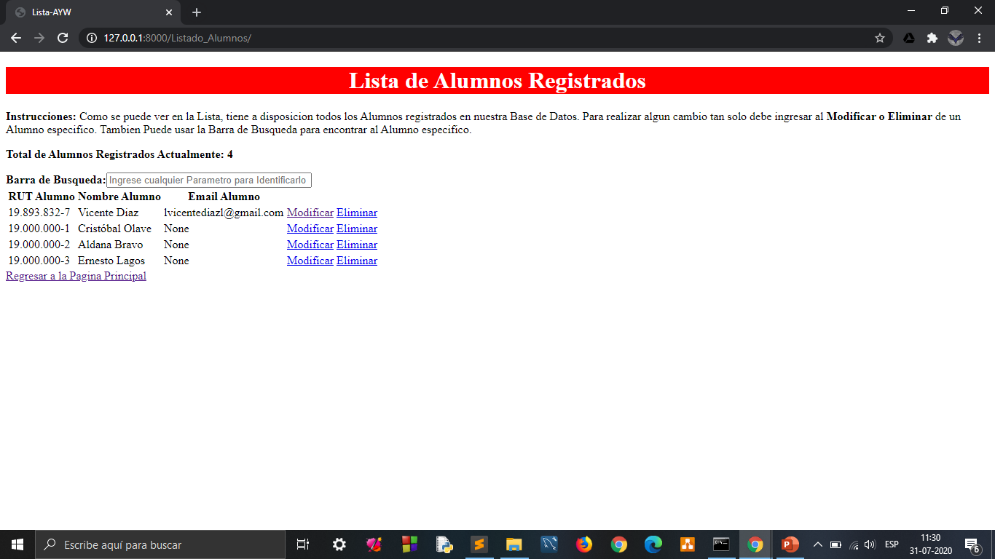
* Operacional, estos deben incluir:
* Monto de compra vigente
* Descuento en base a monto de pago
* Fecha de vencimiento
* Saldo disponible
* Clases tomadas y cobros aplicados
  + 1. CLASE – “Clase”:
* Asistencia por fecha
* Promedio mensual de asistencia
* Promedio mensual de ingreso
* Cantidad y % de tipo de compra
  + 1. CLASE – “Local”:
* Rendimiento general (resumen de ingreso mensual por clase)
* Promedio de ingreso mensual
* Tasa de asistencia mensual
* Nº de alumnos individuales mensual
* Nº de alumnos nuevos por mes
* Ficha de alumnos nuevos unificada mensual
* Cantidad y % de tipo de compra
  + 1. CLASE – “Global”:
* Rendimiento general (resumen de ingreso mensual por local)
* Promedio de ingreso mensual
* Tasa de asistencia mensual
* Nº de alumnos nuevos por mes
  + 1. Niveles de Permiso:

Conforme se va avanzando en el Proyecto, las comunicaciones con el tercero van en aumento para confirmar cambios.

Se nos dio informe sobre los Niveles de Permiso; en este consiste en desarrollar 3 Niveles de Permiso:

* ***Nivel de Profesores:*** Los Usuarios con este Nivel Únicamente tendrán permiso para registrar las Asistencias las Clases de su Local
* ***Nivel de Anfitrión****:* Los Usuario con este Nivel tendrán permitido la Lectura de Datos de cada Sede, sin embargo este no tendrá permitido el Crear, Eliminar, o Modificar cualquier Objeto
* ***Nivel de Director:*** Los Directores tendrán absolutamente todos los Niveles de Permiso posible, por lo que además de leer y registrar asistencias; este tendrá también permitido administrar todo Objeto presente de la Base de Datos, independientemente de la Sede.

En resumen, en algún momento del Sprint 2 o 3, se tendrá que dividir los Permisos de Usuarios, para que además de los que nos están registrados en el Sistema. Los que si están registrados únicamente puedan acceder a Paginas por los cuales tienen permisos como Usuarios.

1. Gestión de Construcción:
   1. Diseño
      1. ***Estético:***

Actualmente nos encontramos en la **Versión v1.2**, esto significa que nos encontramos en la Etapa final del Proyecto, por lo que finalmente se reamplazara el Diseño HTML por un Diseño .Css para que sea visualmente estético para el Cliente, pero que a su vez no afecte con la labor de la Pagina

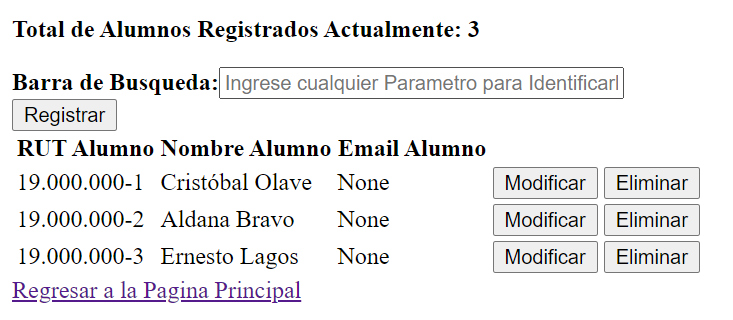
* + 1. Uso:

El Diseño para su uso, está basado en Programación Árbol, es decir, ciertas páginas desplegaran un listado de URL para redireccionarse a otras páginas, de este modo se mantendrá cierto orden de seguimiento, ya sea para avanzar a otras páginas o bien regresar a la Página de inicio.

5.1.1.1 Imagen - Estético

Más específicamente, nuestra página se dividirá en 2 Partes, una será para los que tengan acceso Básico de Usuario los cuales solamente serán capaz de registrar asistencia, mientras que la 2° Parte será para la gestión de datos; para ello este también se dividirá en 2 tipos de Usuarios. El Usuario Anfitrión únicamente será capaz de visualizar los datos por lo que este NO podrá modificar la Base de Datos, el otro tipo de Usuario será el Director, el cual control total de los Datos que se maneja en las Academias, pero este no tendrá acceso para registrar Asistencia.

Dentro de las Páginas en donde se gestiona los Datos se tendrá dentro un numero de opciones específicas que contendrán las Páginas; las opciones más comunes que tendrán las Páginas serán las de Agregación, Modificación, Eliminación y el de retornar a la Página de Inicio. Otras tendrán la capacidad de almacenar lista dentro de Objetos específicos.

* + 1. Estructura:

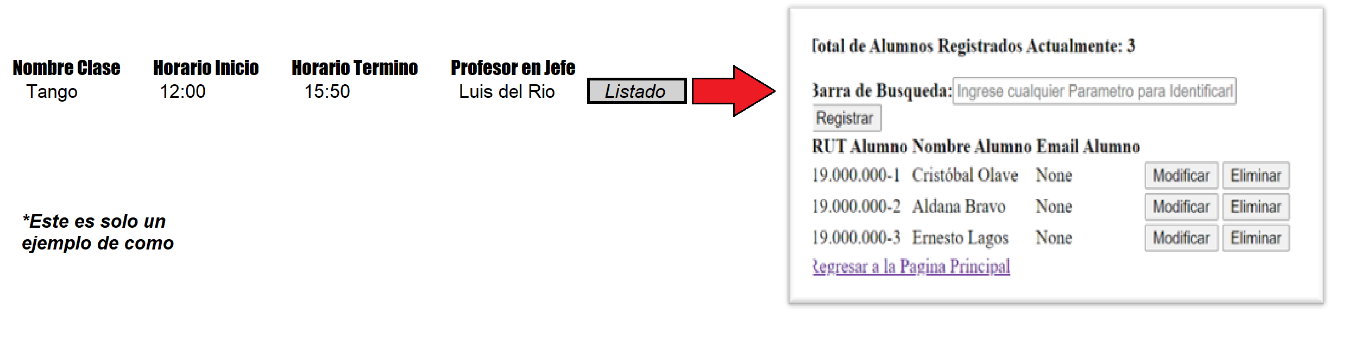
A diferencia del Diseño de Uso de la Aplicación, aquí se demostrara como es la Estructura interna en formato HTML de cada Página.

Para su Estructura, este será construido en formato Tabla (Gracias a la Opción <Table> del Lenguaje de Programación HTML), gracias a ello se podrá desplegar los Listados de los Objetos que se registraran en una Base de Datos externa.

5.1.3.1 Imagen - Estructura

Cada Objeto mostrara en Pantalla un cierto número de atributos específicos que estén dentro de su Modelo (En caso de que muestre otro Atributo, es debido aquel el Objeto se relacionó con el Atributo de otro tipo de Objeto) y que a su vez sean solicitados a través del HTML, al lado de ellos, estos también tendrán una Serie de Botones a su Derecha que les permitirá realizar otras acciones, tales como el de Eliminar el Objeto o Bien Modificarlo. Hay que tomar en cuenta por el momento que una vez realizado las Opciones de “Modificar” o “Eliminar” en el Objeto, este no podrá volver a su estado previo, esto es debido a que no se ha diseñado una Base de Datos que almacene Datos previos que fueron alterados, por lo que se requiere tener mucho cuidado a la Hora de realizar cambios.

Van a existir casos en que ciertos Objetos no solamente requerirán de los botones de “Modificar” o “Eliminar”, tales son los caso de las Clases de “Clase” y “Eventos”, que además de mostrar su Atributos que los distingan de los demás; estos también deben mostrar quienes son los Participantes en dichas Clases o bien otros objetos. Para resolver tal dilema, la Estructura que se planea implementar, es que se implemente un Botón de “Participantes” u otro Boton, el cual al presionarlo este redireccionara a otra Página que mostrara en formato de Tabla, todos los Objetos que están relacionados al Objeto seleccionado (Normalmente, estos de relacionaran mediante algún ID o Nombre Especifico)



5.1.1.1 Imagen - Estético

* Al presionar en el Botón de “Listado” este desplegara en otra Página el Listado (en forma de Tabla) todos los participantes que estén relacionados por algún ID o Nombre especifico.
  + 1. Lenguaje Programación:

Dado que se utilizó Lenguaje DJANGO, este posee un diseño basado en la creación de distintas aplicaciones Django que al juntarse, formaran el programa o proyecto completo. Más adelante en el informe se hablara en mayor detalle. Pero cabe resaltar que gracias a este diseño, es posible realizar futuras actualizaciones, sin perder de vistas el progreso actual.

* 1. Implementación e Integración:

***Misión:*** Construir una Aplicación de gestión de datos, pero que además cuente con otras características que giren en torno en ayudar a las Academias en otras Áreas fundamentales

***Visión:*** Que la aplicación se convierta en una herramienta completa en todos los sentidos, es decir, que las academias usen la Aplicación para completar todas las tareas que son fundamentales y que no requieran del uso de otros Programas o Aplicaciones externas

***Objetivos:*** Que gracias a la aplicación, se logre una mejor administración y por ende, que la academia sea capaz de admitir a una mayor cantidad de Alumnos para incrementar su prestigio y ganancias.

* + 1. Implementación:

Ya completado el Sprint 2, finalmente se está implementando los detalles finales para concluir todo lo relacionado al Proyecto. Por ende la Estructura estaría terminada; por lo que en vez de crear constante Aplicaciones Django, ahora se le dará prioridad en el desarrollar toda función que pueda permitir ejecutar la Pagina.

Para nuestro proyecto se está implementando mediante el uso de Aplicaciones Django, los cuales están divididas con respecto a su Área de Uso, estas son conocidas como ***“Base\_###”***, los cuales ya tienen completado su estructura Base para implementar funciones de complejidad más avanzadas. Dichas aplicaciones son:

***Base\_Inicio:*** Implementa principalmente la página inicial de nuestro proyecto, permitiendo acceder a las otras Aplicaciones ***“Base\_###”***

***Base\_Alumno:*** Implemento las estructura Base de todo lo relacionado a “Alumno”, es decir, implementa sus Atributos básicos, así como las funciones de modificación y eliminación de ese Objeto

***Base\_Profesor:*** Implemento las estructura Base de todo lo relacionado a “Profesor”, es decir, implementa sus Atributos básicos, así como las funciones de modificación y eliminación de ese Objeto

***Base\_Clase:*** Implemento las estructura Base de todo lo relacionado a “Clase”, es decir, implementa sus Atributos básicos. Para el segundo Sprint, se empezara a implementar funciones para registrar asistencia de Alumnos y quienes son los participantes de dichas Clase.

***Base\_Global:*** Implemento las estructura Base de todo lo relacionado a “Global”, es decir, implementa sus Atributos básicos, con el fin de mostrar en formato de Tabla, los Estados Mensuales de las Academias

***Base\_Local:*** Implemento las estructura Base de todo lo relacionado a “Local”, es decir, implementa sus Atributos básicos. En un futuro se planea dividir por Secciones todo los Locales para mostrar su Estado de forma individual

***Base\_User:*** ***:*** Implemento las estructura Base de todo lo relacionado a “User”, es decir, implementa sus Atributos básicos, así como también el de crear Usuarios para el Proyecto. De momento esta Aplicación está en su faceta final para terminar su Estructura

Ya que se está trabajando en formato DJANGO, este puede hacer uso de las “Aplicaciones” para implementar mejoras en dicho Proyecto, sin embargo, el actualizar aplicaciones ya existentes requerirá del reescribir las líneas de código ya establecidas en las aplicaciones antiguas.

* + 1. Integración:

La **Versión v1.2** actualmente se encuentra en faceta final, esto significa que lo único que fata será agregarlo las ultimas funciones que permitan ejecutar el Proyecto sin problemas

* Su **Alcance** está definido por completar oficialmente el Proyecto, estableciendo un método para que pueda almacenar Datos en la Nube.
* La **información** esta identificada en ver como responden las diferentes funciones frente a diversas operaciones o solicitudes, y confirmar que responde de acuerdo a la situación
* Las **Actividades** giran en base a finalizar el Proyecto oficialmente
* Los **Avances** que presenta este Proyecto consisten en resolver todos los fallos o faltas que presento el Sprint 2.
  1. Pruebas de Verificación:

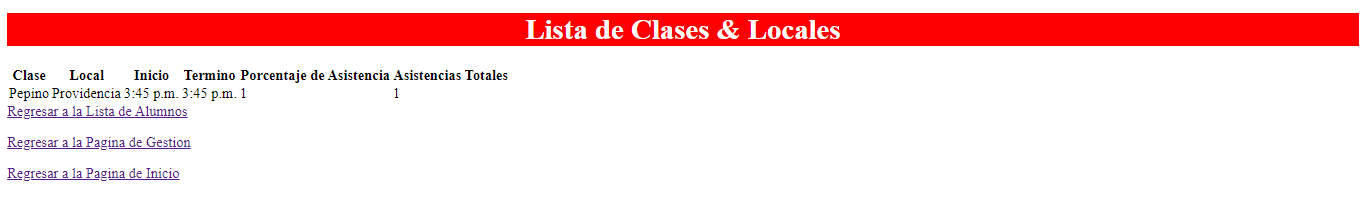
Al ser una de las Ultimas Semanas de Desarrollo, la **Version v1.2.** deberá presentar para su Próximo informe todo avance que se deba resolver esta semana.

Para ello este debe terminar todo lo que consiste el Proyecto para su funcionamiento.

Ya completado la gestión de objetos, así como la gestión de Lista. El Sprint 3 debe resolver todo lo que involucre diseño. css o todo lo que tenga que ver con el establecimiento de su página Web.

Anteriormente se estableció que las Listas son almacenadas en la Base de Datos DB.sqlite3, cosa que deberá resolverse para esta Semana. Dichas lista son relacionadas con otro objeto en especifico mediante el uso de un filtro de búsqueda interno, en base a un atributo que es único en para cada objeto. Gracias a aquello, toda Lista solicitada de un objeto, puede mostrar en Pantalla otros Objetos específicos que guardan cierta relación con el primer objeto; estos además tienen la capacidad de realizar métodos de gestión básico presentes en la interfaces, permitiendo el agregar, modificar, o eliminar dichos Objetos de la Base de Datos.





5.3.1 Imagen – Entregable Simple

* + 1. Logros:
* Establecimiento Total de las estructuras de las Paginas
* Establecimiento de las funciones que implementa
* Establecimiento de la gestión de Lista
  + 1. Fallos de dichos Logros:
* Requiere aun el Implementar los Niveles de Permiso
  + 1. Fallos:
* El implementar un Diseño .Css para darle un Aspecto más estético
* Realizar el establecimiento oficial de la Pagina, esto significa establecer una Base de datos compartida
  + 1. Plan de Testing:

|  |  |  |
| --- | --- | --- |
| Nombre | Descripción | Resultado |
| *Login* | * El Usuario debe registrarse cumpliendo con los requerimientos establecidos | **En desarrollo**  El Usuarios debe establecer también la Residencia a la que pertenece además de ingresar su Nombre y Contraseña.  Este también debe establecer su Nivel de Usuario |
| *Registrar Asistencias* | * Despliega Clases registradas, junto con un filtro de búsqueda * Al seleccionar una Clase, este despliega un Listado de Alumnos perteneciente de dicha Clase * Registra la Asistencia de Alumnos seleccionados | **En desarrollo**  El formulario no está respondiendo a las llamadas de funciones que modifican los Objetos |
| *Lista de Usuario* | * Despliega la Pagina mostrando los Usuarios registrados. * Da la opción de registrar nuevos Usuarios | **En desarrollo**  Si bien despliega a los Usuarios, y realiza registros, este debe permitirnos establecer los Nv. de Permiso |
| *Lista de Alumnos* | * Despliega la Pagina mostrando los Alumnos registrados. * Permite la Administración de dichos Objetos * Despliega el Listado de “Clases & Locales” y “Pagos” perteneciente a un Alumno especifico * Los Listados mencionados permiten la administración de los Objetos específicos de dichos Listados | **Completado**  Aunque este debe separar sus Datos dependiendo de la residencia en la que pertenezca. |
| *Lista de Profesores* | * Despliega la Pagina mostrando los Profesores registrados. * Permite la Administración de dichos Objetos | **Completado**  Aunque este debe separar sus Datos dependiendo de la residencia en la que pertenezca. |
| *Lista de Clases* | * Despliega la Pagina mostrando las Clases registradas. * Permite la Administración de dichos Objetos * Despliega el Listado de “Participantes” y “Ingresos” perteneciente a una Clase especifica * Los Listados mencionados permiten la administración de los Objetos específicos de dichos Listados | **Completado**  Aunque este debe separar sus Datos dependiendo de la residencia en la que pertenezca. |
| *Lista de Eventos* | * Despliega la Pagina mostrando los Eventos registrados. * Permite la Administración de dichos Objetos * Despliega el Listado de “Participante pertenecientes a un Evento especifico * Los Listados mencionados permiten la administración de los Objetos específicos de dichos Listados | **Completado**  Aunque este debe separar sus Datos dependiendo de la residencia en la que pertenezca. |
| *Lista de Academias* | * Despliega la Pagina mostrando los ingresos totales de la Academia perteneciente | **Completado**  Aunque este debe separar sus Datos dependiendo de la residencia en la que pertenezca. |
| *Estado de la Academia* | * Despliega la Pagina mostrando los ingresos totales mensuales de la Academia en general | **Completado** |

5.3.1.1 Tabla – Pruebas Registradas

* 1. Pruebas de Validación:

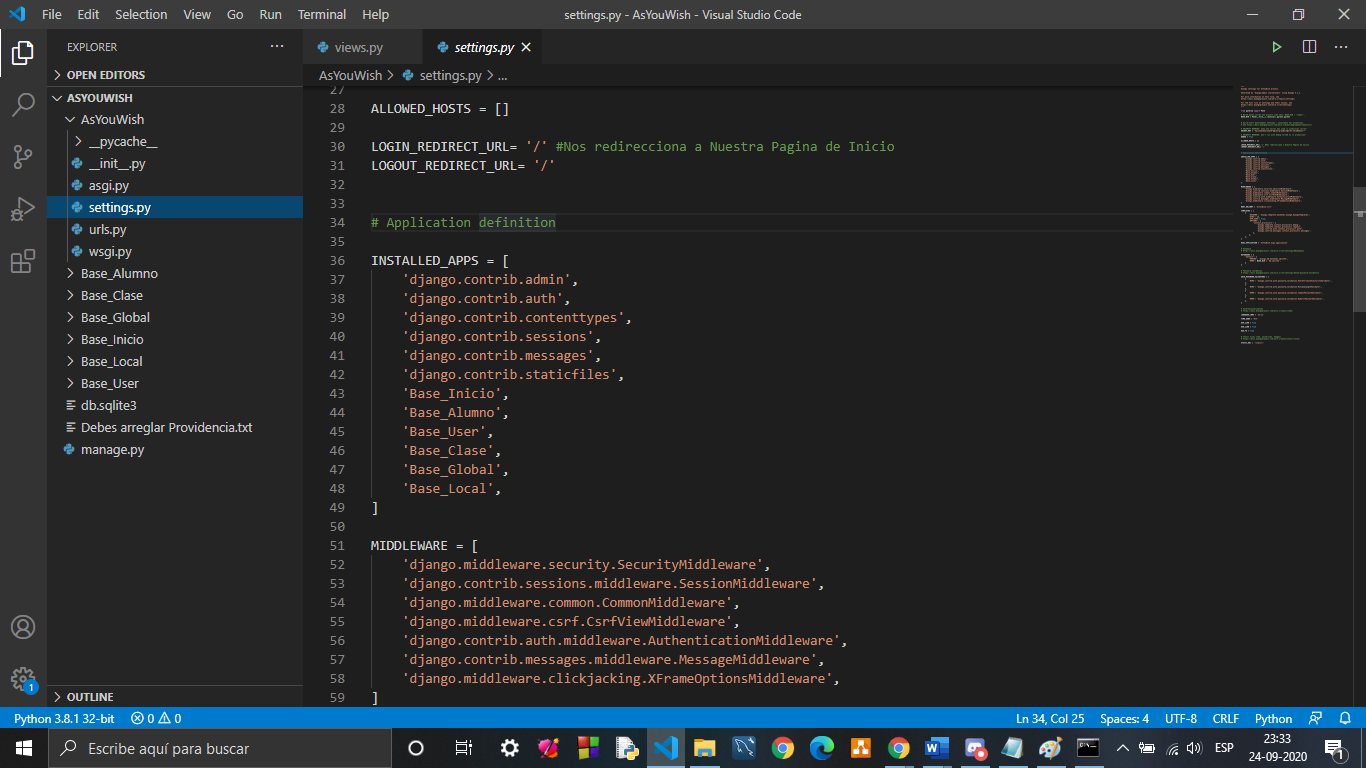
De momento, únicamente se nos ha solicitado, y por ende empezado a desarrollar un Software el cual pueda reunir los datos de cada sede de As You Wish, ser almacenados y ordenados para una óptima gestión y administración en las futuras actividades de eventos y clases que se abren a lo largo del año, sin dejar de lado tener un mejor estudio del comportamiento de la circulación de gente para planificaciones de nuevas mejoras.

Aun se continua en contacto con el tercero que nos comunica cualquier cambio que quiera presentar la Academia. Semanalmente se envían informes para confirmar que la dirección del proyecto es igual a la que tenían previsto.

1. Gestión Deployment:

Con forme se va avanzo en el Proyecto, este implemento diversas funciones y objetos en el programa. Por ello, este debió ser ordenada de forma correcta para tener una buena comprensión del código.

Finalmente se esta desarrollando las partes finales del proyecto, que consisten en los elementos que se mencionan en el Sprint 3.



6.1 Imagen - Setting

* Aun se continua con el Programa Django, pues este nos permite clasificar las funciones gracias a la creación de múltiples Aplicaciones Django, que almacenan funciones específicas, además de que estas pueden ser importadas a otras aplicaciones para que estas hagan uso de sus funciones o datos

1. [ACTUALIZADO] Gestión Proyectos:

Actualmente el Proyecto se encuentra en el ***3° Sprint – Implementar Diseño***

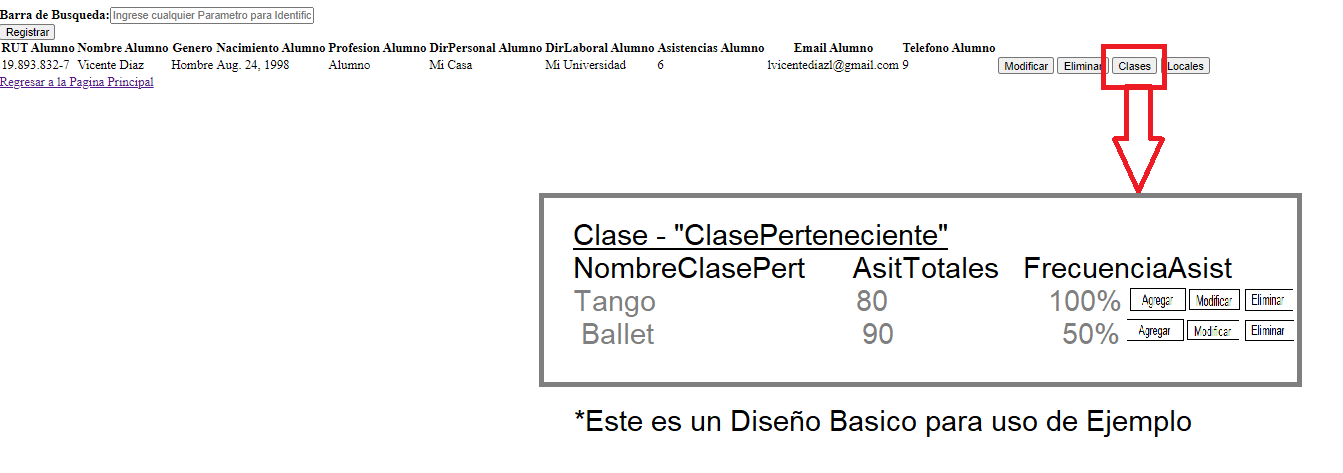
***3° Sprint - Implementar Diseño***

* *Ejecución inmediata de la Aplicación [Prioridad = 3 /Complejidad = 5]*
* *Establecer Nivel de Permiso de Usuario [Prioridad = 3 /Complejidad = 15]*
* *Conexión a Base de Datos en la Nube [Prioridad = 3 /Complejidad = 5]*
* *Establecer Diseño .Css para todas las Páginas [Prioridad = 3 /Complejidad = 5]*
* *Escribir/Establecer Manual de Usuario [Prioridad = 3 /Complejidad = 1]*
  1. Planificación:

Para completar las funciones finales del Proyecto. Este debe ser capaz de ejecutarse inmediatamente en la Red; es por ello que se le debe establecer un URL de Pagina Web para que todo público pueda tener acceso a él, pero estableciendo ciertos niveles de seguridad para mantener seguros los Datos; pero a su vez establecer un Diseño .Css o Bootstrap para darle cierto estilo propio a la Pagina.

* 1. Diseño de Estructura principal

Ya establecido la Estructura en código del Programa, lo único que queda por concluir, es el establecimiento de los Servidores Online, así como el establecer su Diseño .Css.



7.2.1 Imagen – Estructura General

* 1. Diseño de Estructura de Código

Ya completado la estructura base, lo único en que se puede trabajar en el de resolver los pequeños fallos que presenta la Pagina. Pero más allá de eso no se puede definir una Estructura Clave que aún se deba implementar; ya que todo se terminó de establecer.

Actualmente se esta planificando el cómo conseguir un Dominio de URL, así como el establecimiento de Servidor y Base de Datos en la Nube

* 1. [ACTUALIZADO] Actividades y sus Responsables
     1. Establecer funciones finales de Ingreso

**Responsable a Cargo:** Vicente Díaz

* El Encargado debe completar el establecimiento de Dominio de URL de la Pagina “Asyouwish”, así como el establecer la Base de Datos en la nube para lograr una Base de Datos compartida.

Para ello, primero debe confirmar con nuestro cliente un Dominio de URL, (Actualmente se esta esperando respuesta). para luego finalmente ser utilizada para todo el Publico.

* + 1. Establecer funcionaes finales de Usuarios

**Responsable a Cargo:** Cristobal Olave

* Se debe completar la Pagina Log In del Programa, así como el establecer la funciones finales para agregar y eliminar a un Nuevo Usuario, estableciendo su Nivel de Permiso
* Una vez completado todo, deberá establecer los niveles de permiso en todas las páginas del Proyecto
  + 1. Establecer .Css

**Responsable a Cargo:** Ernestos Lagos

* El encargado debe terminar de diseñar el Bootstrap .Css para el diseño de la Pagina
  + 1. Establecer .Css

**Responsable a Cargo:** Aldana Bravo

* El encargado debe terminar de diseñar el Bootstrap .Css para el diseño de la Pagina
  + 1. Monitoreo & Control:

***Página de Inicio***

|  |  |  |
| --- | --- | --- |
| *Función* | *Estado* | *Prioridad* |
| Levantar Página HTML | COMPLETADO | 1 |
| Implementar Enlace para Cerrar Sesión | COMPLETADO | 2 |
| Implementar URL “Lista de Usuarios” | COMPLETADO | 1 |
| Implementar URL “Lista de Alumnos” | COMPLETADO | 1 |
| Implementar URL “Profesores” | COMPLETADO | 1 |
| Implementar URL “Lista de Clases | COMPLETADO | 1 |
| Implementar URL “Lista de Locales” | COMPLETADO | 1 |
| Implementar URL “Estado Global” | COMPLETADO | 1 |
| Implementar Diseño .CSS | En desarrollo | 3 |

7.5.5.1 Tabla – Inicio

***Clase –“Alumno”***

|  |  |  |
| --- | --- | --- |
| *Función* | *Estado* | *Prioridad* |
| Levantar Página HTML | COMPLETADO | 1 |
| Implementar URL para retornar a la Página Principal | COMPLETADO | 2 |
| Implementar Tabla HTML mostrando Datos de sus Atributos | COMPLETADO | 1 |
| Implementar Botón de Función “Registrar”, su respectiva Página | COMPLETADO | 1 |
| Implementar Botón de Función “Modificar” | COMPLETADO | 1 |
| Implementar Botón de Función “Eliminar” | COMPLETADO | 1 |
| Implementar Botón de Función “Clases & Locales” | COMPLETADO | 2 |
| Resultados de Pagos | En desarrollo | 2 |
| Implementar Filtro de Búsqueda | COMPLETADO | 2 |
| Implementar Diseño .CSS | En desarrollo | 3 |

7.5.5.1 Tabla – Alumno

***Clase – “Profesor”***

|  |  |  |
| --- | --- | --- |
| *Función* | *Estado* | *Prioridad* |
| Levantar Página HTML | COMPLETADO | 2 |
| Implementar URL para retornar a la Página Principal | COMPLETADO | 2 |
| Implementar Tabla HTML mostrando Datos de sus Atributos | COMPLETADO | 2 |
| Implementar Botón de Función “Modificar” | COMPLETADO | 2 |
| Implementar Botón de Función “Eliminar” | COMPLETADO | 2 |
| Implementar Barra de Búsqueda | COMPLETADO | 2 |
| Implementar Diseño .CSS | En desarrollo | 3 |

7.5.5.2 Tabla - Profesor

***Clase – “Usuario”***

|  |  |  |
| --- | --- | --- |
| *Función* | *Estado* | *Prioridad* |
| Levantar Página HTML | COMPLETADO | 1 |
| Implementar URL para retornar a la Página Principal | COMPLETADO | 2 |
| Implementar Tabla HTML mostrando Datos de sus Atributos | COMPLETADO | 2 |
| Implementar Botón de Función “Registrar”, su respectiva Página | En desarrollo | 2 |
| Implementar Botón de Función “Eliminar” | En desarrollo | 2 |
| Implementar Diseño .CSS | En desarrollo | 3 |
| Implementar Niveles de Permiso | En desarrollo | 3 |

7.5.5.3 Tabla – Usuario

***Clase – “Clase”***

|  |  |  |
| --- | --- | --- |
| *Función* | *Estado* | *Prioridad* |
| Levantar Página HTML | COMPLETADO | 1 |
| Implementar URL para retornar a la Página Principal | COMPLETADO | 2 |
| Implementar Tabla HTML mostrando Datos de sus Atributos | COMPLETADO | 2 |
| Implementar Botón de Función “Registrar”, su respectiva Página | COMPLETADO | 2 |
| Implementar Botón de Función “Participantes”, su respectiva Página | COMPLETADO | 2 |
| Implementar Botón de Función “Modificar”, su respectiva Página | COMPLETADO | 2 |
| Implementar Botón de Función “Eliminar” | COMPLETADO | 2 |
| Implementar Filtro de Búsqueda | COMPLETADO | 2 |
| Implementar Diseño .CSS | En desarrollo | 3 |
| Implementar Registro de Asistencias | COMPLETADO | 1 |

7.5.5.4 Tabla - Clase

***Clase – “Local”***

|  |  |  |
| --- | --- | --- |
| *Función* | *Estado* | *Prioridad* |
| Levantar Página HTML | COMPLETADO | 1 |
| Implementar URL para retornar a la Página Principal | COMPLETADO | 1 |
| Implementar Tabla HTML mostrando Datos de sus Atributos | COMPLETADO | 1 |
| Resultados de Compras | COMPLETADO | 2 |
| Implementar funciones que actualicen en tiempo real los Atributos | COMPLETADO | 2 |
| Implementar Diseño .CSS | En desarrollo | 3 |

7.5.5.5 Tabla - Local

***Clase – “Global”***

|  |  |  |
| --- | --- | --- |
| *Función* | *Estado* | *Prioridad* |
| Levantar Página HTML | COMPLETADO | 1 |
| Implementar URL para retornar a la Página Principal | COMPLETADO | 1 |
| Implementar Tabla HTML mostrando Datos de sus Atributos | COMPLETADO | 1 |
| Implementar funciones que actualicen en tiempo real los Atributos | COMPLETADO | 2 |
| Implementar Diseño .CSS | En desarrollo | 3 |

7.5.5.6 Tabla - Global

***Funciones que no se pueden Clasificar***

|  |  |  |
| --- | --- | --- |
| *Función* | *Estado* | *Prioridad* |
| Crear Base de Datos | COMPLETADO | 1 |
| Implementar Servidores | En desarrollo | 3 |
| Ejecución inmediata de la Aplicación | En desarrollo | 3 |

7.5.5.7 Tabla – No Clasificadas

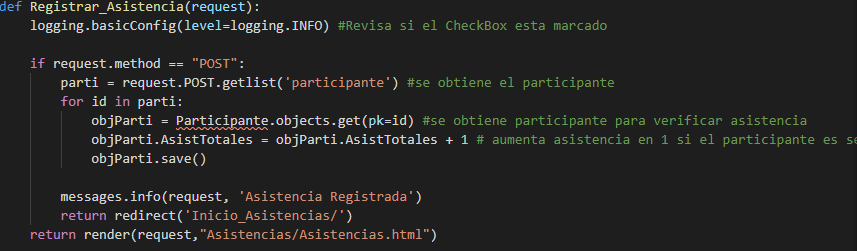
* 1. Entregables
     1. Cristobal Olave

Actualmente no se a logrado establecer niveles de permiso para los Usuarios, se esta teniendo problemas para definir los tres permiso que fueron solicitados. “Básico”, “Anfitrión” & “Director”.

Se están intentando diversos métodos y funciones pero aun se sigue sin poder establecer los Niveles de Permiso

* + 1. Aldana Bravo

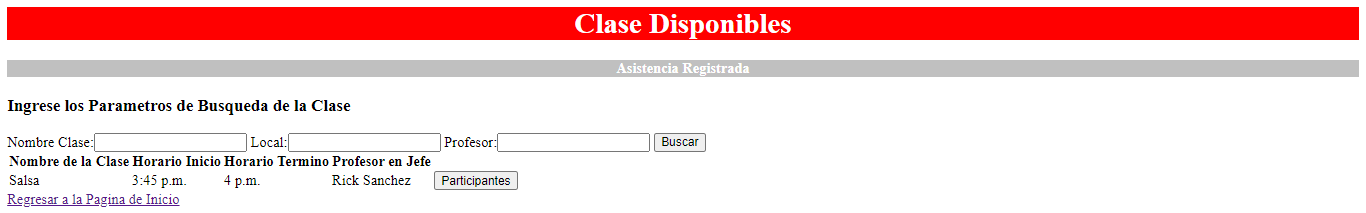
El Desarrollo para registrar la Asistencias de Alumno específicos fue completada. De esta forma cada vez que se registre la asistencia, el Alumno que tenia marcado su *CheckBox* solicitara que su Asistencia aumente en 1. Por ende, ahora se esta brindando apoyo para completar el .Css



7.5.2.1. Imagen – Codigo Registrar\_Asistencia

* Aquí se presenta el Codigo, gracias al uso de *logging.basicconfig*, el programa es capaz de identificar a aquellos participantes que tienen marcado su *Checkbox*

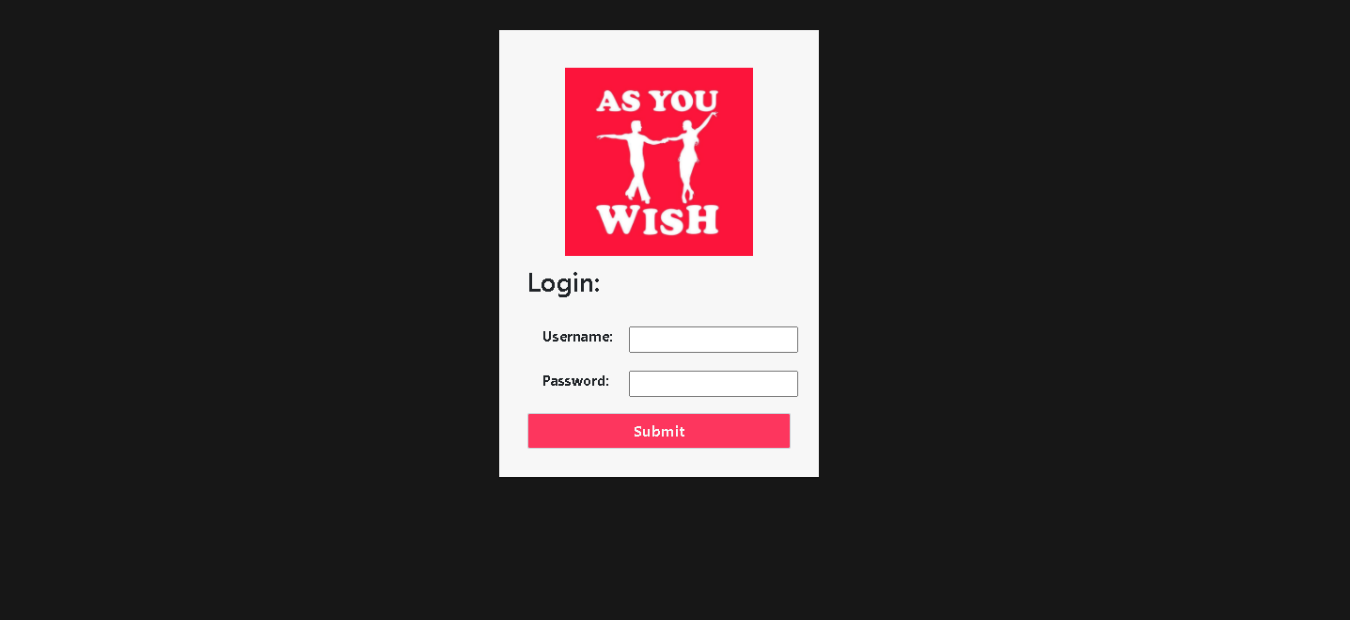




7.5.2.2 Imagen – Registro HTML

* Por ende registrar la Asistencia, se enviara un Mensaje Flash que notifique que la Asistencia ya fue registrada exitosamente.
  + 1. Nicolas Lagos

Ya completado el Eliminar Participantes al eliminar a una Clase o Evento, ahora se esta trabajando en implementar un Diseño .Css al Codigo. Por ende principalmente se esta trabajando en más concretamente en modificar el HTML. Por ende era fundamental que toda función haya sido completada.



7.5.3.1 Imagen – Diseño .Css

* Como se puede ver, aquí hay un ejemplo del cómo se vería el Login de nuestra Pagina Web. Más específicamente, se esta buscando implementar un patrón de diseño en Rojo & Negro
  + 1. Vicente Díaz

La implementación de crear los Ingresos que recibe un Local o Academia ya fue completado con éxito. Con ello tambien se mejoro el formulario de pago que realiza un Alumno.



7.5.4.1 Imagen – Nuevo Form Pago

* Este al seleccionar el Local, Año y Mes. El codigo revisara si el Pago ya existe con los Atributos que permiten identificar los Pagos previos, si estos no existen, se creara uno Nuevo. De lo contrario modificara el que ya esta previamente creado.



7.5.4.2 Imagen – Pagos Local

* Como se puede ver, los ingresos totales seran equivalentes a la suma total de todos los apgos que se han realizado en ese Local



7.5.4.3 Imagen – Pago Estado Mensual

* Ademas de eso, este tambien crea o modifica objetos para tener un registro del Estado Mensual (y Anual) que lleva la academia en total. Es decir, los Ingresos Totales de “Estado Mensual” se acumulan si tienen en comun el Mes y Año en que se realizo el Pago.

Cabe tambien mencionar que al eliminar un Pago, este tambien lo resta de los Pagos que reciben los Locales y la Academia para no tener un registro equivocado de los Pagos realizados



7.5.4.4 Imagen – Pago Eliminado

* Como se puede ver al Eliminar los Pagos de Diciembre 1998 & Septiembre de 2020, estos se restan para actualizar los Datos (Este es solo un Ejemplo, si el Pago llega a 0 el Objeto es Eliminado de forma inmediata)

Por ultimo, al combinar el trabajo actual de todos tambien se pulieron los codigos:

***“Eliminar Particiante”:*** Al eliminar un Participante de una Clase, este reduce en 1 el Registro de un Genero en especifico para tener los registros actualizados

***“Verificar Alumno”:*** Para registrar un nuevo participantes, el programa primero confirma si dicho alumno esta registrado en la Clase “Alumno” antes de registrarlo como Participante de alguna Clase o Evento

***“Registrar Asistencia”:*** Principalmente, se elimino ciertas lineas de codigo que tienen 0 relevancia de lo que debe mostrar. Y de esa forma no estar abrumado por toda la linea de codigo que estaba presente