***[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 – 6° Semana

16 de Octubre de 2020

*Contenidos Generales:*

[1 Introducción: 4](#_Toc53769965)

[2 Descripción Proyecto: 4](#_Toc53769966)

[3 [ACTUALIZADO] Mapa de Historias: 5](#_Toc53769967)

[4 [ACTUALIZADO] Gestión de Requerimientos: 6](#_Toc53769968)

[4.1 Alcance 6](#_Toc53769969)

[4.2 Calidad 7](#_Toc53769970)

[4.3 Costo 7](#_Toc53769971)

[4.4 Tiempo 7](#_Toc53769972)

[4.5 Req.Especiales 8](#_Toc53769973)

[5 Gestión de Construcción: 9](#_Toc53769974)

[5.1 Diseño 9](#_Toc53769975)

[5.2 Implementación e Integración: 11](#_Toc53769976)

[5.3 [ACTUALIZADO] Pruebas de Verificación: 13](#_Toc53769977)

[5.4 Pruebas de Validación: 18](#_Toc53769978)

[6 Gestión Deployment: 19](#_Toc53769979)

[7 [ACTUALIZADO] Gestión Proyectos: 20](#_Toc53769980)

[7.1 Planificación: 20](#_Toc53769981)

[7.2 Diagrama de Clases 21](#_Toc53769982)

[7.3 Diseño de Estructura principal 22](#_Toc53769983)

[7.4 Diseño de Estructura de Código 23](#_Toc53769984)

[7.5 Requisitos 23](#_Toc53769985)

[7.6 [ACTUALIZADO] Actividades y sus Responsables 24](#_Toc53769986)

[7.7 Entregables 29](#_Toc53769987)

*Contenido de Imágenes:*

[3.1 Imagen – Mapa de Historia 5](#_Toc53769988)

[4.4.1 Imagen - GANTT 7](#_Toc53769989)

[5.1.1.1 Imagen - Estético 9](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53769990)

[5.1.3.1 Imagen - Estructura 10](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53769991)

[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-%206°Semana.docx#_Toc53769992)

[5.3.1 Imagen – Entregable Simple 14](#_Toc53769993)

[6.1 Imagen - Setting 19](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53769994)

[7.2.1 Imagen – Diagrama de Clases 21](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53769995)

[7.3.1 Imagen – Diseño Estructural 22](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53769996)

[7.3.2 Imagen - Ejemplo 22](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53769997)

[7.4.1 Imagen - Código 23](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53769998)

[7.2.1 Imagen – Diagrama de Clases 25](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53769999)

[7.7.1.1 Imagen – PagoAlumno 29](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770000)

[7.7.2.1 Imagen – HTML Aldana 30](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770001)

[7.7.3.1 Imagen – LocalesAsistidos 31](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770002)

[7.7.3.2 Imagen – HTML Ernesto 31](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770003)

[7.7.4.1 Imagen – Participante 32](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770004)

[7.7.4.2 Imagen – HTML Participante 32](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770005)

[7.7.4.3 Imagen – Formulario Participante 33](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770006)

[7.7.4.5 Imagen – Resultado 33](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770007)

[7.7.4.4 Imagen – Widgets 33](file:///C:\Users\Pokev\Desktop\Segundo%20Semestre\Ing.Software%20II\Documentos%20de%20Ing.Software\Informes%20Semanales\Informe%20Avance%20-%206°Semana.docx#_Toc53770008)

*Contenido de Tablas:*

[3.1 Tabla – Mapa de Historia 6](#_Toc53770009)

[5.3.1.1 Tabla – Pruebas Registradas 18](#_Toc53770010)

[7.5.5.1 Tabla – Inicio 25](#_Toc53770011)

[7.5.5.1 Tabla – Alumno 26](#_Toc53770012)

[7.5.5.2 Tabla - Profesor 27](#_Toc53770013)

[7.5.5.3 Tabla – Usuario 27](#_Toc53770014)

[7.5.5.4 Tabla - Clase 28](#_Toc53770015)

[7.5.5.5 Tabla - Local 28](#_Toc53770016)

[7.5.5.6 Tabla - Global 28](#_Toc53770017)

[7.5.5.7 Tabla – No Clasificadas 29](#_Toc53770018)

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. [ACTUALIZADO] 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. [ACTUALIZADO] 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 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 esta 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 v1.2**, por el momento, la fecha más estimada para lograr un avance notable seria el **26 de Oct.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:

**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

**CLASE – *“Clase”***

* Asistencia por fecha
* Promedio mensual de asistencia
* Promedio mensual de ingreso
* Cantidad y % de tipo de compra

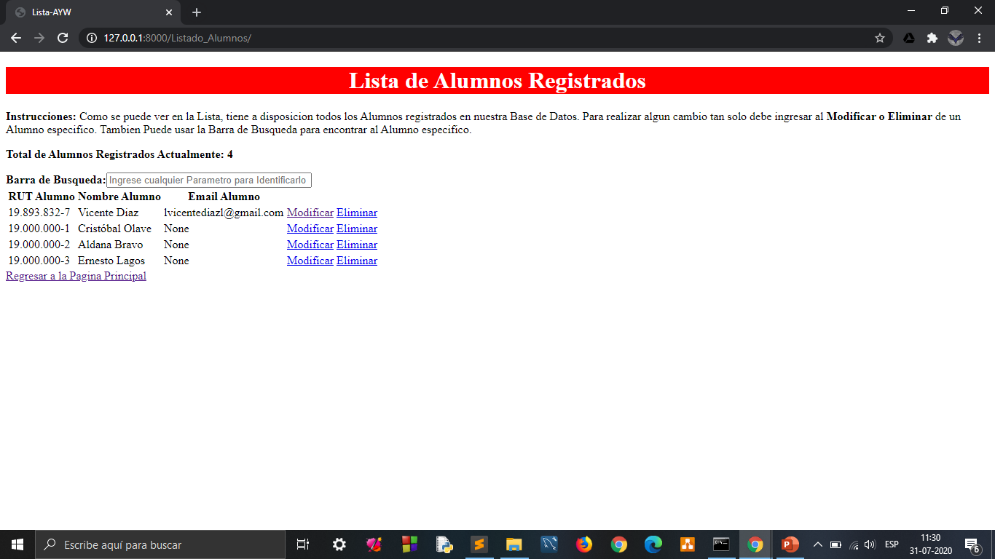
**CLASE – *“Local”***

* 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

**CLASE – *“Global”***

* Rendimiento general (resumen de ingreso mensual por local)
* Promedio de ingreso mensual
* Tasa de asistencia mensual
* Nº de alumnos nuevos por mes

Más allá de eso, no se solicitó algo otro requerimiento especial en específico.

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

Actualmente nos encontramos en la **Versión v1.1**, por lo que se le dio prioridad al desarrollo de las Páginas principales del Proyecto. Dejando el desarrollo de **.CSS** con prioridad mínima.

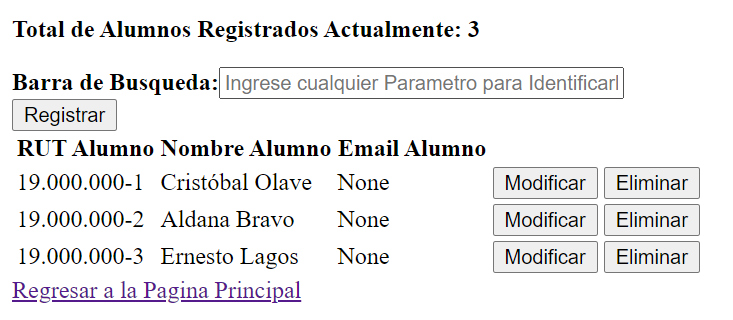
Por ello, dado que se decidió dejar el Diseño con prioridad mínima, este aun permanecerá en fase de espera, por consiguiente, únicamente se implementó un diseño haciendo uso de las funciones Limitadas que otorga los Templates del Lenguaje HTML para una mejor visualización de la construcción de la Pagina.

5.1.1.1 Imagen - Estético

* + 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. De momento este diseño aún sigue en desarrollo por lo que se espera futuras actualizaciones en esta versión

Más específicamente, se podrá acceder a otras Páginas de la Aplicación desde la Página de Inicio (Siempre y Cuando este se haya iniciado Sesión de Usuario), a partir de ahí se tendrá la lista de URL de diferentes Áreas de la Academia [Usuario, Profesor, Alumno, Clases, Local, Global, Eventos, entre otras]. Dentro de cada Página se tendrá dentro un numero de opciones específicas que contendrán las Páginas; por el momento, las opciones más comunes que tendrán las Páginas serán las de Agregación, Modificación, Eliminación y el de retornar a la Página de Inicio

* + 1. Estructura:

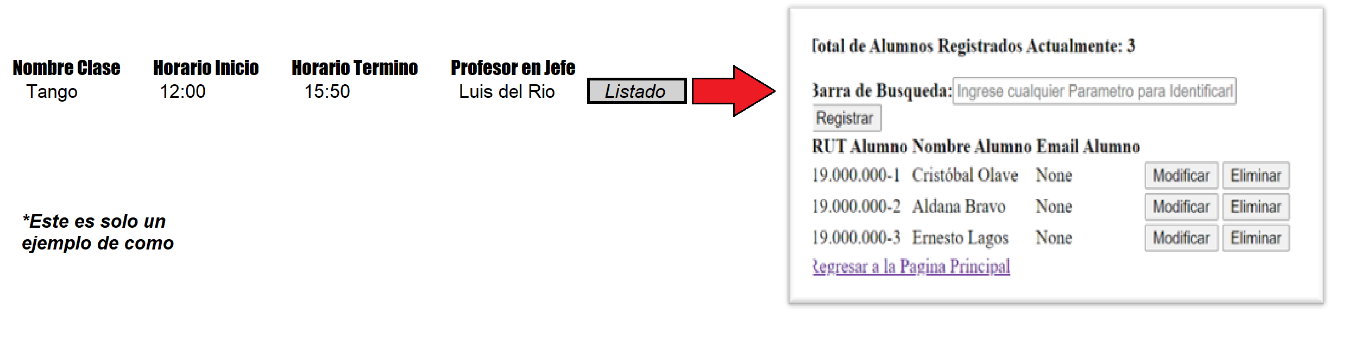
A diferencia del Diseño de Uso de la Aplicación, aquí se demostrara como es la Estructura interna en formato HTML de cada Página.

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 registran en la Base de Datos de DB.Sqlite3.

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. Para resolver tal dilema, la Estructura que se planea implementar, es que se implemente un Botón de “Participantes”, 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)



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.

5.1.1.1 Imagen - Estético

Es posible que conforme se avance en el Proyecto, se implementen más Botones que complementen las otras funciones del Proyecto

* + 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 1, ahora se está trabajando en concluir todo lo relacionado a Listas, por lo que en vez de crear constante Aplicaciones Django, ahora se le dará prioridad en el desarrollar nuevos Archivos y Carpetas que complementen todas las funciones relacionada a Listas de Objetos, dividiendo el trabajo en función de cada característica principal del proyecto.

Actualmente 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

Gracias a que previamente se dejó encargados para investigar sobre las Funciones de Listas, se implementara un Plan de desarrollo para lograr con el cometido de crear Lograr las Listas en Base a lo que se investigó.

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 v1.1** actualmente se encuentra en faceta de actualización, esto significa que conforme avance el proyecto se espera futuras líneas de código en aplicaciones Django ya existente, pero es poco probable la creación de nuevas Aplicaciones Django.

* Su **Alcance** está definido por completar todas las funciones principales del Proyecto, una vez completado eso se le dará prioridad el completar todo lo relacionado a la implementación de servidores y base de datos online
* 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 sobre el crear funciones que permitan la creación de Listas, así como el sus Funciones
* Los **Avances** que presenta este Proyecto es la definición de Páginas principales que servirán como punto de referencia para avanzar en el trabajao, es decir, a medida que se avance en el Proyecto, se espera nuevas líneas de código que desarrollen mejor todas las Páginas.
  1. [ACTUALIZADO] Pruebas de Verificación:

Al ser **Versión v1.1**, la definición de Páginas Estructurales base están definidas. Por lo que cada uno puede cumplir la función de **Registrar y Archivar Datos de distintos Objetos**, sin embargo, este flaquea en la capacidad de administrarlas en mayor nivel; por lo que en esta versión se empezara en trabajar sobre todo lo relacionado a funciones complejas de la Aplicación, más concretamente, **todo lo relacionado a Listas.**

De momento no se ha ninguna fallo critico en el desarrollo del Proyecto, por lo que funciona de manera operativa (aunque de forma simple) y en proceso de Actualización, así como las funciones que apoyan o guardan cierta relación con dicha función (Eliminar, Modificar, Ver y buscar diversos Objetos en formato Tabla).

Toda Página creada para la Aplicación funciona de manera óptima también, es decir, ninguna falla al solicitar un redireccionamiento de página, así como las solicitudes y registros de formulario que se programaron.

La Estructura Base del Proyecto se encuentra en buenas condiciones, sin embargo únicamente cumple con requisitos básicos y ninguna función de alta complejidad.

5.3.1 Imagen – Entregable Simple

**Logros:**

* Establecimiento correcto de la Estructura Base del Proyecto
* Muestra de forma correcta los Objetos en sus respectivas Áreas
* Registro y gestión de Objeto

**Fallos de dichos Logros:**

* Se encuentra con una complejidad sumamente baja al flaquear de funciones de alto calibre
* El mostrar un Listado de Objetos específicos aún sigue en faceta de desarrollo (Este estaría completo para el 2°Sprint)

**Fallos:**

De momento no se ha conseguido algún fallo

* + 1. Plan de Testing:

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Descripción** | **Resultado** |
| **Login** | El Usuario debe ingresar al Sistema con Nombre & Contraseña, y selección de Residencia | En desarrollo.  A pesar de que nos solicita Nombre & Contraseña, aun falta implementar los Niveles de Permiso, asi como el seleccioanr la Residencia |
| **LogOut** | El Usuario debe cerrar su Sesion cuando seleccione el URL “Cerrar Sesion” | Completado |
| [Lista Alumnos](http://127.0.0.1:8000/Listado_Alumnos/) | URL que redirecciona a su HTML | Completado |
| [Crear Usuario](http://127.0.0.1:8000/CrearUsuarios/) | URL que redirecciona a su HTML | Completado |
| [Lista Profesores](http://127.0.0.1:8000/Listado_Profesor/) | URL que redirecciona a su HTML | Completado |
| [Lista Academias](http://127.0.0.1:8000/Academias/) | URL que redirecciona a su HTML | Completado |
| [Lista Clases](http://127.0.0.1:8000/Listado_Clase/) | URL que redirecciona a su HTML | Completado |
| [Lista Eventos](http://127.0.0.1:8000/Listado_Evento/) | URL que redirecciona a su HTML | Completado |
| [Estado de la Academia](http://127.0.0.1:8000/Registro_Global/) | URL que redirecciona a su HTML | Completado |
| [HTML] [Lista Alumnos](http://127.0.0.1:8000/Listado_Alumnos/) | HTML que despliega tanto Atributos como Objetos de su Clase perteneciente | Completado |
| [HTML] [Crear Usuario](http://127.0.0.1:8000/CrearUsuarios/) | HTML que despliega formulario para registrar Usuario Nuevo | Completado |
| [HTML] [Lista Profesores](http://127.0.0.1:8000/Listado_Profesor/) | HTML que despliega tanto Atributos como Objetos de su Clase perteneciente | Completado |
| [HTML] [Lista Academias](http://127.0.0.1:8000/Academias/) | HTML que despliega Listado de URL para acceder al Registro de Otras Sedes | En Desarrollo.  Se requiere implementar nuevos URL, asi como sus respectivos HTML |
| [HTML] [Lista Clases](http://127.0.0.1:8000/Listado_Clase/) | HTML que despliega tanto Atributos como Objetos de su Clase perteneciente | Completado |
| [HTML] [Lista Eventos](http://127.0.0.1:8000/Listado_Evento/) | HTML que despliega tanto Atributos como Objetos de su Clase perteneciente | Completado |
| [HTML] [Estado de la Academia](http://127.0.0.1:8000/Registro_Global/) | HTML que despliega tanto Atributos como Objetos de su Clase perteneciente | En Desarrollo.  Requiere implementar Funciones que registren en tiempo real los Datos |
| [Forms] [Lista Alumnos](http://127.0.0.1:8000/Listado_Alumnos/) | Al presionar el Boton “Registrar” o “Modificar”, se despliega un Formulario que debe registrar los Atributos pertenecientes de un Nuevo Objeto | Completado.  No acepta Atributos que sean de Tipo de Dato Diferente |
| [Forms] [Crear Usuario](http://127.0.0.1:8000/CrearUsuarios/) | Al presionar el Boton “Registrar” o “Modificar”, se despliega un Formulario que debe registrar los Atributos pertenecientes de un Nuevo Objeto | Completado.  No acepta Atributos que sean de Tipo de Dato Diferente.  No Acepta Tipo de Datos que no cumplan con los Requerimientos de tener un Minimo de 8 Caracteres |
| [Forms] [Lista Profesores](http://127.0.0.1:8000/Listado_Profesor/) | Al presionar el Boton “Registrar” o “Modificar”, se despliega un Formulario que debe registrar los Atributos pertenecientes de un Nuevo Objeto | Completado.  No acepta Atributos que sean de Tipo de Dato Diferente |
| [Forms] [Lista Clases](http://127.0.0.1:8000/Listado_Clase/) | Al presionar el Boton “Registrar” o “Modificar”, se despliega un Formulario que debe registrar los Atributos pertenecientes de un Nuevo Objeto | Completado.  No acepta Atributos que sean de Tipo de Dato Diferente |
| *[Forms]* [*Lista Eventos*](http://127.0.0.1:8000/Listado_Evento/) | Al presionar el Boton “Registrar” o “Modificar”, se despliega un Formulario que debe registrar los Atributos pertenecientes de un Nuevo Objeto | Completado.  No acepta Atributos que sean de Tipo de Dato Diferente |
| *[Delete]* [*Lista Alumnos*](http://127.0.0.1:8000/Listado_Alumnos/) | Al presionar el Boton “Eliminar” se borra el Objeto seleccionado de la Base de Datos | Completado.  Requiere implementar un Metodo que confirme la solicitud de eliminación |
| *[Delete]* [*Lista Profesores*](http://127.0.0.1:8000/Listado_Profesor/) | Al presionar el Boton “Eliminar” se borra el Objeto seleccionado de la Base de Datos | Completado.  Requiere implementar un Metodo que confirme la solicitud de eliminación |
| *[Delete]* [*Lista Clases*](http://127.0.0.1:8000/Listado_Clase/) | Al presionar el Boton “Eliminar” se borra el Objeto seleccionado de la Base de Datos | Completado.  Requiere implementar un Metodo que confirme la solicitud de eliminación |
| *[Delete]* [*Lista Eventos*](http://127.0.0.1:8000/Listado_Evento/) | Al presionar el Boton “Eliminar” se borra el Objeto seleccionado de la Base de Datos | Completado.  Requiere implementar un Metodo que confirme la solicitud de eliminación |
| *[Clases]* [*Lista Alumnos*](http://127.0.0.1:8000/Listado_Alumnos/) | Al presionar el Boton “Clases” redirecciona a otro HTML que despliega una Tabla “Clases\_Perteneciente” | En Desarrollo.  Se requiere implementar el HTML |
| *[Locales]* [*Lista Alumnos*](http://127.0.0.1:8000/Listado_Alumnos/) | Al presionar el Boton “Locales” redirecciona a otro HTML que despliega una Tabla “Locales\_Perteneciente” | En Desarrollo.  Se requiere implementar el HTML |
| *[Participantes]* [*Lista Clases*](http://127.0.0.1:8000/Listado_Clase/) | Al presionar el Boton “Participantes” redirecciona a otro HTML que despliega una Tabla “Participantes\_Clases” | En Desarrollo.  Requiere implementar Funciones para mejor gestion |
| *[Ingresos]* [*Lista Clases*](http://127.0.0.1:8000/Listado_Clase/) | Al presionar el Boton “Eliminar” se borra el Objeto seleccionado de la Base de Datos | Completado.  Requiere implementar un Metodo que confirme la solicitud de eliminación |
| *[Participantes]* [*Lista Eventos*](http://127.0.0.1:8000/Listado_Evento/) | Al presionar el Boton “Participantes” redirecciona a otro HTML que despliega una Tabla “Participantes\_Evento” | En Desarrollo.  Se requiere implementar el HTML |
| *[Search]* [*Lista Alumnos*](http://127.0.0.1:8000/Listado_Alumnos/) | Filtro de busqueda | En Desarrollo.  Solo puede realizar busquedas en base al Atributo “Nombre" |
| *[Search]* [*Lista Profesores*](http://127.0.0.1:8000/Listado_Profesor/) | Filtro de busqueda | En Desarrollo.  Solo puede realizar busquedas en base al Atributo “Nombre" |
| *[Search]* [*Lista Clases*](http://127.0.0.1:8000/Listado_Clase/) | Filtro de busqueda | En Desarrollo.  Solo puede realizar busquedas en base al Atributo “Nombre" |
| *[Search]* [*Lista Eventos*](http://127.0.0.1:8000/Listado_Evento/) | Filtro de busqueda | En Desarrollo.  Solo puede realizar busquedas en base al Atributo “Nombre" |

5.3.1.1 Tabla – Pruebas Registradas

* 1. Pruebas de Validación:

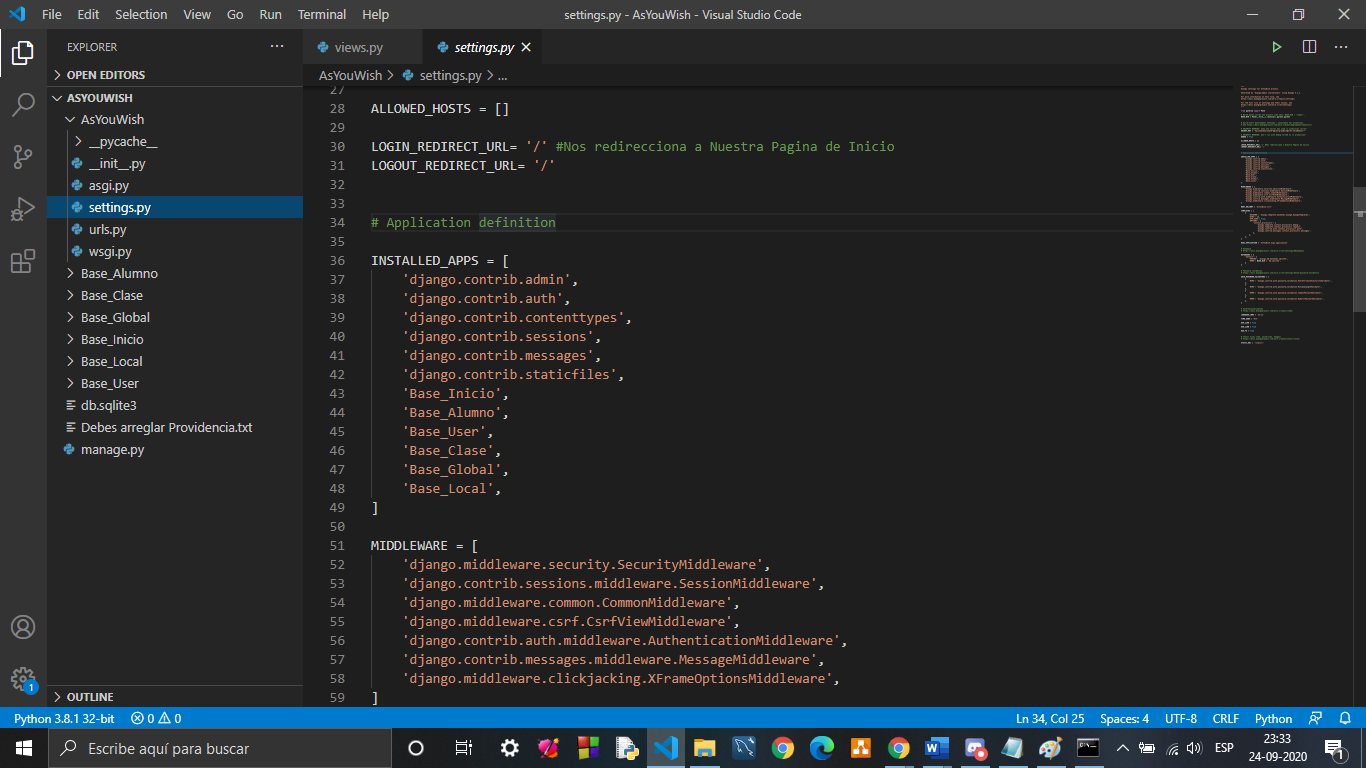
De momento, únicamente se nos ha solicitado, y por ende empezado a desarrollar un Software el cual pueda reunir los datos de cada sede de As You Wish, ser almacenados y ordenados para una óptima gestión y administración en las futuras actividades de eventos y clases que se abren a lo largo del año, sin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

Dado que ahora nos encontramos en el 2°Sprint, se espera mayores pruebas que verifiquen el correcto desarrollo del proyecto.

1. Gestión Deployment:

Con forme se va avanzando en el Proyecto, este implementara cada vez más objetos y funciones dentro del programa, por lo que es fundamental clasificarlos con tal de evitar confusión a la hora de crear o actualizar.

Por ello, a diferencia de la primera versión, este fue desarrollada o más bien ordenada para que su entendimiento de donde esta cada función sea más pulida. La Máxima prioridad del Proyecto es el correcto manejo de los Objetos de la Clase “Alumno” debido a que este es la solicitud principal que nos pidió nuestro Cliente



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

6.1 Imagen - Setting

1. [ACTUALIZADO] Gestión Proyectos:

Actualmente el Proyecto se encuentra en el ***2° Sprint – Funciones de Listado***

***2° Sprint - Funciones de Listado***

* Completar las Funciones que permitan todas las Características de un Listado:
  + Capacidad de Crear/Eliminar/Modificar Listas ***[En Desarrollo]***
  + Agregar/Eliminar/Ver Objetos REGISTRADOS de un Lista en Específico, especificando qué Atributos deben ser visibles (Ej:Nombre, Rut, etc) ***[En Desarrollo]***
  + Registrar Asistencia para los Alumnos ***[En Desarrollo]***
  + Clasificar/Registrar/Ver diferentes Listas (Es decir, ser capaz de Archivar y Administrar todas las Listas de Forma Sencilla)
* Conectarlos/Relacionarlos a los URLs de las otras Páginas ***[En Desarrollo]***
  1. Planificación:

Una vez completado el Sprint 1 del Proyecto, en donde se ponía especial énfasis en el desarrollo de la Estructura Base de la Aplicación. En el Sprint 2 se pondrá énfasis en el Desarrollo de las funciones complejas que involucran cada Objeto registrado, con el objetivo de implementar Listas para que permitan un mayor Orden para los Datos.

Para ello la Planificación, se dividirá en partes iguales para cada integrante del Grupo, en donde cada uno cumple con una Tarea en específico.

En estas Semanas del 2° Sprint se enfocara principalmente en el Desarrollo de Listas para cada Objeto, más específicamente en el levantamiento de Páginas HTML. Para ello una gran variedad de Objetos tendrá asignado una variedad de Listas de otros Objetos unidos mediante “relaciones” para ampliar aún más sus Datos. Estos además tendrán incluidos una variedad de funciones que permiten modificar tantos los Atributos Propios como los de otros Objetos(Más adelante se especificara en detalle para cada objeto), por ende es fundamental estar en comunicación constante para saber los Nombres de los Atributos y variables, y de esta forma evitar cualquier confusión en adelante.

Para ello, primero se debe definir un Plan de Desarrollo que permita el correcto direccionamiento del desarrollo del Código y definir un Diagrama de Clases que defina que Clases estarán relacionadas con esa Clase.

* 1. Diagrama de Clases

Como saben un Diagrama de Clases tiene la tarea principal de definir las relaciones de una clase con otra**,** de esta forma Objeto de dicha clase pueden mantener una “relación” que permite interactuar con los Atributos de otros Objetos de una Clase distinta.

A continuación se mostrara una Versión Actualizada del Diagrama de Clase “As You Wish” para visualizar dichas relaciones.

Diagrama

Descripción generada automáticamente

7.2.1 Imagen – Diagrama de Clases

Como pueden ver cada Clase guarda algún tipo de relación con otra, por lo que estas interactúan de forma directa entre sí.

En base a la imagen anterior, cada Objeto tendrá una “relación” con un Objeto perteneciente a otras Clase, permitiendo un intercambio de atributos y más adelante una interacción mediante funciones.

* 1. Diseño de Estructura principal

A continuación se mostrara un ejemplo de la próxima Estructura de Diseño.

Para el Objeto de Clase “Alumno” se tiene 4 Botones que redirigen hacia otras Paginas

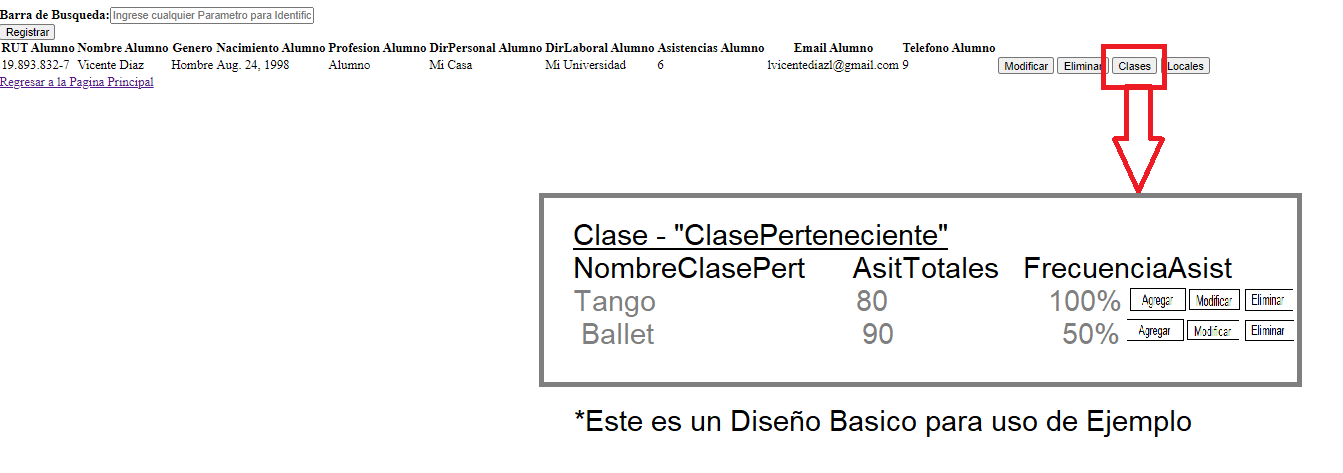


7.3.1 Imagen – Diseño Estructural

*Modificar & Eliminar:* Llevan a Páginas que ya está consideradas completadas en bajo nivel, por lo que por el momento **no requieren de modificación alguna**.

*Clases & Locales*: Son un ejemplo de las Páginas que cada integrante debe desarrollar para el 2° Sprint. Cada un Página debe mostrar una Lista de Tabla <Table> de todos los Objetos de una Nueva Clase que se debe desarrollar. Dicha Tabla mostrara nuevos Atributos que tienen la posibilidad de afectar los Atributos de Otros Objetos.

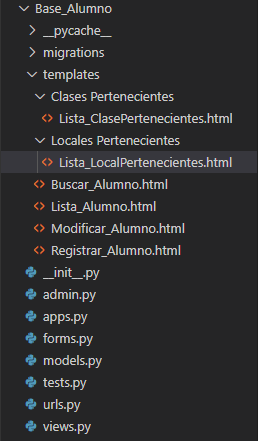
***Ej:***



7.3.2 Imagen - Ejemplo

Como se puede ver, su Diseña únicamente involucra mostrar en una Tabla todos los objetos registrados que guardan alguna relación con el Objeto seleccionado. Por ello consiste en que cada Objeto tendrá a su Derecha unos Botones que redireccionaran hacia otras páginas, mostrando otro Listado de Objeto que guardan cierta relación con el Objeto anterior perteneciente.

* 1. Diseño de Estructura de Código

Una vez definido su estructura de interfaz, lo siguiente es el explicar cómo debe ser su estructura a Nivel de Código, para ello vamos hacer uso de **“Base\_Alumno”** como ejemplo.

Como pueden ver la imagen a su Derecha, diseñar dicha Estructura va involucrar el crear nuevos Folder y Files para el Código, de una forma en que puedan ser fácilmente reconocidos y organizados.

Para **“Base\_Alumno”** un ejemplo será el desarrollar nuevos HTML para crear nuevas paginas; por lo que involucraría modificar también views.py & urls.py (Hay que tener en cuenta que esos .py se ven directamente influenciados por dichos HTML). Y al ser listado de Objetos nuevos también involucrara el modificar el model.py (Más adelante se detallaran los Atributos que deben tener principalmente).

Por el momento el diseñar funciones estará en espera hasta que se complete el Diseño de las nuevas Páginas

* 1. Requisitos

1. Esta estrictamente prohibido el Modificar cualquier otra Aplicación **“Base\_###”** que no se le fue asignada

7.4.1 Imagen - Código

1. De momento únicamente se concentrara el levantar Páginas que muestren un Listado de Objetos mediante relación; las funciones se detallaran para la proxima Planificación
2. Está prohibido el Diseñar nuevas Aplicaciones Django
3. Está prohibido el crear Atributos que no guarden algún relación con lo planificado
4. Estos deben tener los Nombres exactos que se describen en el Documento
5. Cada Página debe tener un Botón de “Agregar”, “Modificar”, “Eliminar”; los cuales son parecidos al de las Páginas principales, así como sus funciones
   1. [ACTUALIZADO] Actividades y sus Responsables
      1. Crear Clase “ParticipanteClase” en models.py

**Responsable a Cargo:** Vicente Díaz

* **Base a la que pertenece:** *“Base\_Clase”*
* **Clase con el que se relaciona:** Clase *“Alumno”* perteneciente a *“Base\_Alumno”*

La creación de esta Clase involucra la capacidad de mostrar cada Alumno que este registrado en una Clases en Especifico, para ello su HTML debe ser capaz de llamar a funciones que permitan registrar nuevos Alumnos, así como también la capacidad de eliminar y mostrar la cantidad de Asistencias que tiene el Alumno registrado

* + 1. Crear Clase “LocalesAsistidos” en models.py

**Responsable a Cargo:** Ernesto Lagos

* **Base a la que pertenece:** *“Base\_Alumno”*
* **Clase con el que se relaciona:** Clase *“Alumno”* perteneciente a *“Base\_Alumno”*

Al igual que con el contenido “7.5.1 Crear Clase “ParticipanteClase” en models.py”, este debe ser capaz de mostrar, registrar, y eliminar Objetos que muestren a que Locales asiste el Alumno actualmente.

* + 1. Crear Clase “PagosAlumno” en models.py

**Responsable a Cargo:** Cristobal Olave

* **Base a la que pertenece:** *“Base\_Alumno”*
* **Clase con el que se relaciona:** Clase *“Alumno”* perteneciente a *“Base\_Alumno”*

Esta Clase debe crear y eliminar Objetos que demuestren los pagos o ingresos que realiza el Alumno. Pero a la vez este debe tener funciones que tenga como parámetros los Ingresos de cada Objeto registrado.

* + 1. Crear Página “ParticipanteEvento”

**Responsable a Cargo:** Aldana Bravo

* **Base a la que pertenece:** *“Base\_Evento”*
* **Clase con el que se relaciona:** Clase *“Evento”* perteneciente a *“Base\_Evento”*

Al igual que con el contenido “7.5.1 Crear Clase “ParticipanteClase” en models.py”, este debe ser capaz de mostrar, registrar, y eliminar Objetos que muestren los Participantes de un Evento en especifico.

7.2.1 Imagen – Diagrama de Clases

* + 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 espera | 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 Pertenecientes”, con % de Asistencias | En espera | 2 |
| Implementar Botón de Función “Locales a los que Asiste” | En espera | 2 |
| Resultados de Compras | En espera | 2 |
| Implementar Barra de Búsqueda | COMPLETADO | 2 |
| Implementar Diseño .CSS | En espera | 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 Botón de Función “Clases Pertenecientes” | En espera | 2 |
| Implementar Botón de Función “Locales a los que Asiste” | En espera | 2 |
| Implementar Barra de Búsqueda | COMPLETADO | 2 |
| Implementar Diseño .CSS | En espera | 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 | En desarrollo | 2 |
| Implementar Botón de Función “Registrar”, su respectiva Página | En desarrollo | 2 |
| Implementar Botón de Función “Modificar”, su respectiva Página | En desarrollo | 2 |
| Implementar Botón de Función “Eliminar” | En desarrollo | 2 |
| Implementar Barra de Búsqueda | En espera | 2 |
| Implementar Diseño .CSS | En espera | 3 |
| Implementar Niveles de Permiso | En espera | 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 | En espera | 2 |
| Implementar Botón de Función “Listado de Participantes”, su respectiva Página | En espera | 2 |
| Implementar Botón de Función “Modificar”, su respectiva Página | En espera | 2 |
| Implementar Botón de Función “Eliminar” | COMPLETADO | 2 |
| Implementar Barra de Búsqueda | COMPLETADO | 2 |
| Implementar Diseño .CSS | En espera | 3 |
| Implementar Registro de Asistencias | En espera | 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 |
| Implementar Botón de Función “Clases Disponibles”, con % de Asistencias | En espera | 2 |
| Resultados de Compras | En espera | 2 |
| Implementar Barra de Búsqueda | En desarrollo | 2 |
| Implementar funciones que actualicen en tiempo real los Atributos | En desarrollo | 2 |
| Implementar Diseño .CSS | En espera | 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 | En desarrollo | 2 |
| Implementar Diseño .CSS | En espera | 3 |

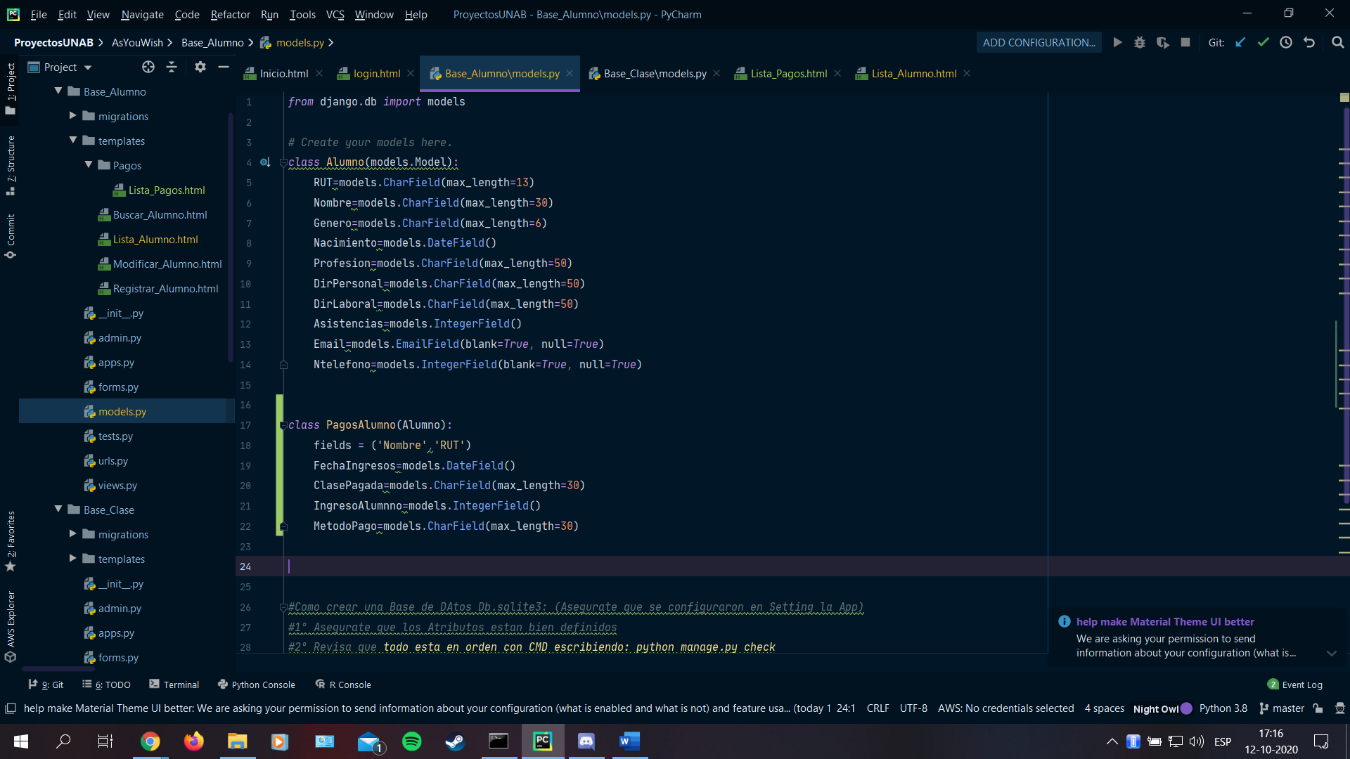
7.5.5.6 Tabla - Global

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

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

7.5.5.7 Tabla – No Clasificadas

* 1. Entregables
     1. Cristobal Olave

Se ha creado exitosamente la clase de PagoAlumno, la cual consiste en los datos financieros del alumno en si. Despues, se creó un botón de pagos en Lista\_Alumnos, la cual aún está en desarrollo.

7.7.1.1 Imagen – PagoAlumno

* + 1. Aldana Bravo

Se pidió realizar la vista de participantes a los eventos, indicando su nombre, el local al cual asisten y nombre de la clase. Información a la que se debe acceder mediante un botón en lista de eventos.

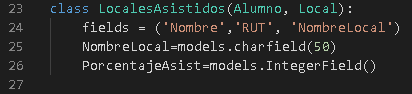
**EN PRIMERA INSTANCIA SE TIENE EL HTML DE LOS PARTICIPANTES A EVENTOS, EN EL CUAL SE BUSCA VISUALIZAR SOLAMENTE EL NOMBRE, EL LOCAL Y LA CLASE A LA QUE CADA ALUMNO ASISTE.**

**Texto

Descripción generada automáticamente**

7.7.2.1 Imagen – HTML Aldana

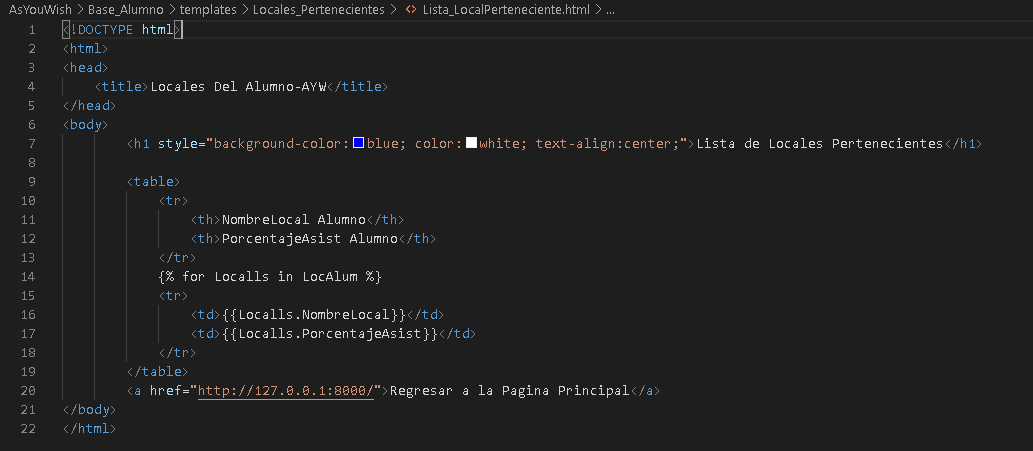
* + 1. Nicolas Lagos

Se agrego la clase LocalesAsistidos en “AsYouWish\Base\_Alumno\models.py”

7.7.3.1 Imagen – LocalesAsistidos

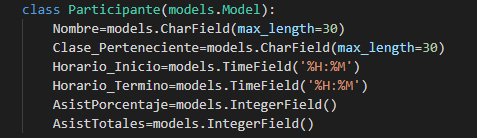
Se creó el html en donde se mostrarán el NombreLocal y el PorcentajeAsist en “AsYouWish\Base\_Alumno\templates\Locales\_Pertenecientes\Lista\_LocalPerteneciente.html”

7.7.3.2 Imagen – HTML Ernesto



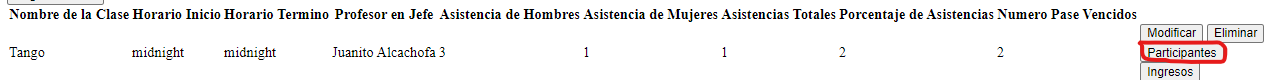
* + 1. Vicente Díaz

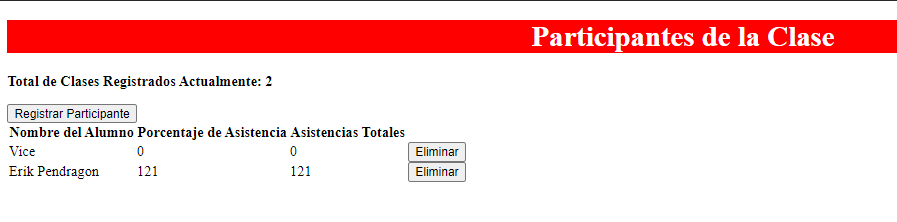
Al encargarse de la creación de “ParticipanteClase”, lo primario era la creación de dicha Clase la cual ya se completó y se realizó la creación de su propia Tabla en la Base de Datos de Sqlite3; definiendo los atributos relevantes que servirán en un futuro para próximas actividades.



7.7.4.1 Imagen – Participante

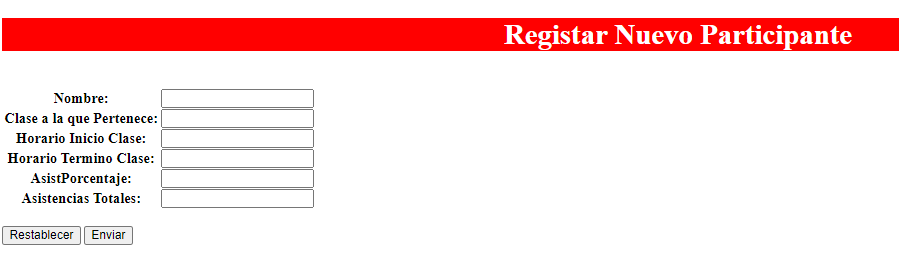
Una vez definido la Clase, este ahora debía ser mostrada en un HTML en formato de Tabla mostrando únicamente los Alumnos que pertenecen a la Clase. Para ello se empleo funciones de filtro que toman como parámetro la “Clase\_Perteneciente” que consiste en que Clase pertenece los Alumno registrados. Al realizarse correctamente, se mostrara en Pantalla todos los Objetos registrados que pertenezcan a la Clase seleccionada mediante el uso del Botón “Participantes” ignorando todos los Objetos que no guardan relación alguna.





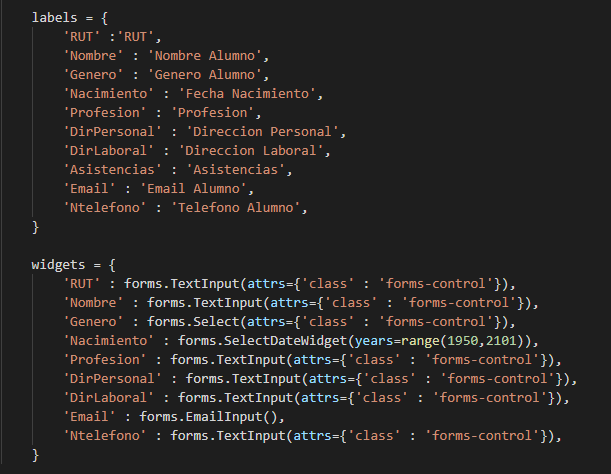
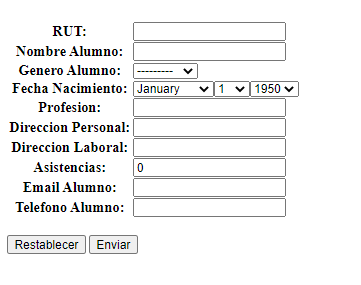
7.7.4.2 Imagen – HTML Participante

También se a logrado el crear un formulario que permite registrar nuevos Alumno, sin embargo este aun requiere de nuevas mejoras debido a que únicamente es capaz de registrar Objetos nuevos, pero no verifica que dicho Objeto sean ya un Alumno existente o bien que este registrado en la Tabla que muestra todos los Alumnos registrados.

Es decir, cuando se registra un Nuevo Objeto, el Sistema no verifica que este ya existe, además de que tampoco cuenta con funciones que permitan una mejor gestión, como por ejemplo el de registrar las Asistencia de forma manual o automática. 

7.7.4.3 Imagen – Formulario Participante

Por ultimo, el programa esta siendo nuevamente ordenado implementando Widgets y Label que le den un Estilo más refinado a cada formulario ya creado. Para la proxima Semana, ya deberian tener implementado todos los Widgets que entreguen una mejor Estilo para el Formulario.



7.7.4.5 Imagen – Resultado

7.7.4.4 Imagen – Widgets