

Tecnicatura Universitaria en Programación

Programación II

Trabajo Práctico Integrador

Consigna

Desarrollar un sistema de Campus Virtual para alumnos y profesores, aplicando los conceptos aprendidos en la materia.

Se deberá entregar el código en un .zip y además el mismo deberá encontrarse en un repositorio de GitHub. Todos los integrantes del equipo deben colaborar en el repositorio. No se podrán incluir cambios en el repositorio pasada la fecha de entrega final.

Se habilitará en el CVG para realizar las entregas en la fecha indicada para su comisión.

La primera entrega es obligatoria para todos los alumnos, tanto para regularizar cómo para acceder a la aprobación directa. La entrega final es optativa, pero de carácter obligatoria si se desea acceder a la aprobación directa.

Los grupos son de 2 o 3 integrantes.

Metodología de evaluación

Se tendrá en cuenta para la nota final

- El uso de git para el trabajo colaborativo
- El correcto uso de Python
- Validaciones extras a las mencionadas que consideren oportunas
- Mensajes extras a los mencionados que consideren oportunos
- Adecuación a PEP8
- Defensa del TPI en grupo
- Calidad del código entregado

Para la entrega final se tendrá en cuenta el informe solicitado.

Instancias de corrección

Si las entregas realizadas en equipo no alcanzan la nota de aprobación, 6, se les otorgará un plazo para que reentreguen el trabajo práctico integrador.

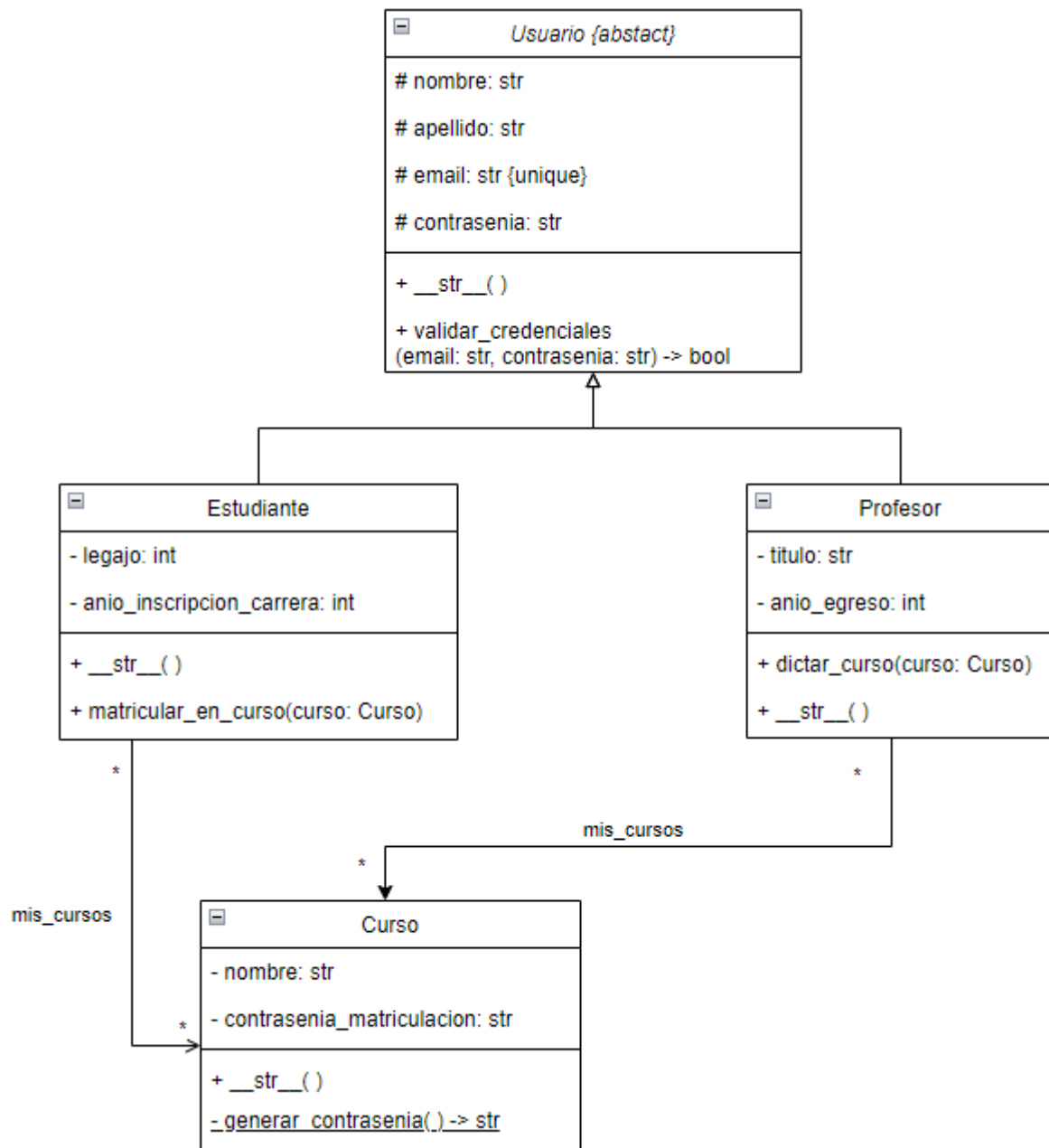
Primera entrega

Se deberá entregar en un documento .pdf la siguiente información:

- Título: Trabajo Práctico Integrador
- Entrega: Primera
- Comisión: XTUPX
- Integrantes: (nombre - legajo)
- Repositorio: Link al repositorio con el trabajo práctico

Además se deberá entregar un .zip con el código.

Codificar el siguiente diagrama UML en Python:



Se debe respetar los siguientes requerimientos:

El sistema cuenta con algunos alumnos y profesores dados de alta. También se cuenta con una carrera dada de alta. No se cuentan con cursos dados de alta al iniciar el sistema.

Al ingresar al sistema se debe mostrar un menú en bucle con las siguientes opciones:

1. [Ingresar cómo alumno](#)
2. [Ingresar cómo profesor](#)
3. [Ver cursos](#)
4. [Salir](#)

Cuando se selecciona la opción 1 se ingresa el email y la password. Primero se busca con el email en la lista de alumnos registrados si el mismo existe.

Si existe se valida que el email y contraseña sean correctos, si son correctos se da acceso al sistema. En caso contrario se muestra un cartel de error de ingreso.

Si el alumno no existe con ese correo se debe mostrar un mensaje diciendo que debe darse de alta en alumnado.

Cuando se selecciona la opción 2 se ingresa el email y la password. Primero se busca con el email en la lista de profesores registrados si el mismo existe.

Si existe se valida que el email y contraseña sean correctos, si son correctos se da acceso al sistema. En caso contrario se muestra un cartel de error de ingreso.

Si el profesor no existe con ese correo se debe mostrar un mensaje diciendo que debe darse de alta en alumnado.

Cuando se selecciona la opción 3, se debe mostrar una lista de todos los cursos del Campus Virtual ordenados alfabéticamente como sigue:

Materia: Ingles I	Carrera: Tecnicatura Universitaria en Programación
Materia: Ingles II	Carrera: Tecnicatura Universitaria en Programación
Materia: Laboratorio I	Carrera: Tecnicatura Universitaria en Programación
Materia: Laboratorio II	Carrera: Tecnicatura Universitaria en Programación
Materia: Programación I	Carrera: Tecnicatura Universitaria en Programación
Materia: Programación II	Carrera: Tecnicatura Universitaria en Programación
...	

Sub menú alumno (cuando se ingresa cómo alumno exitosamente):

1. [Matricularse a un curso](#)
2. [Ver curso](#)
3. [Volver al menú principal](#)

Cuando se selecciona la opción 1, se debe mostrar una lista de todos los cursos del Campus Virtual como sigue:

- 1 [Programación I](#)
- 2 [Programación II](#)
- 3 [Laboratorio II](#)
- 4 [Ingles I](#)
- 5 [Ingles II](#)
- ...

Se valida que se ingrese una opción de las mostradas (1,2,3,4,5,...) y se debe validar que el alumno no se encuentre ya matriculado en dicho curso, es decir que el curso no se encuentra en su lista de mi_cursos. Además se valida que la contraseña de matriculación ingresada sea la del curso. Si se cumplen todas las validaciones se agrega el curso a los cursos del alumno en la lista mi_cursos. En caso contrario se muestran mensajes correspondientes.

Cuando se selecciona la opción 2, se muestra una lista de los cursos que el alumno se encuentra matriculado cómo sigue...

- 1 Programación I
- 2 Programación II
- 3 Laboratorio II

...

Se valida que se ingrese una opción de las mostradas (1,2,3,...) y se recupera el curso y se muestra el nombre del curso.

Nombre: Programación I

Sub menú profesor (cuando se ingresa cómo profesor exitosamente):

- 1. Dictar curso
- 2. Ver curso
- 3. Volver al menú principal

Cuando se selecciona la opción 1, se dará de alta un curso con el nombre del curso ingresado. La contraseña de matriculación se genera automáticamente. Luego de creado el curso el mismo se agrega a la lista mis_cursos del profesor y se deja disponible en el sistema para la matriculación de los alumnos.

Se muestra un mensaje de éxito y todos los datos del curso dado de alta...

Nombre: Programación I

Contraseña: kKj10

Cuando se selecciona la opción 2, se muestra una lista de los cursos que el profesor posee cómo sigue...

- 1 Programación I
- 2 Programación II

...

Se valida que se ingrese una opción de las mostradas (1,2,...) y se muestra el nombre del curso y la contraseña de matriculación, por ejemplo...

Nombre: Programación I

Contraseña: kKj10

Entrega Final

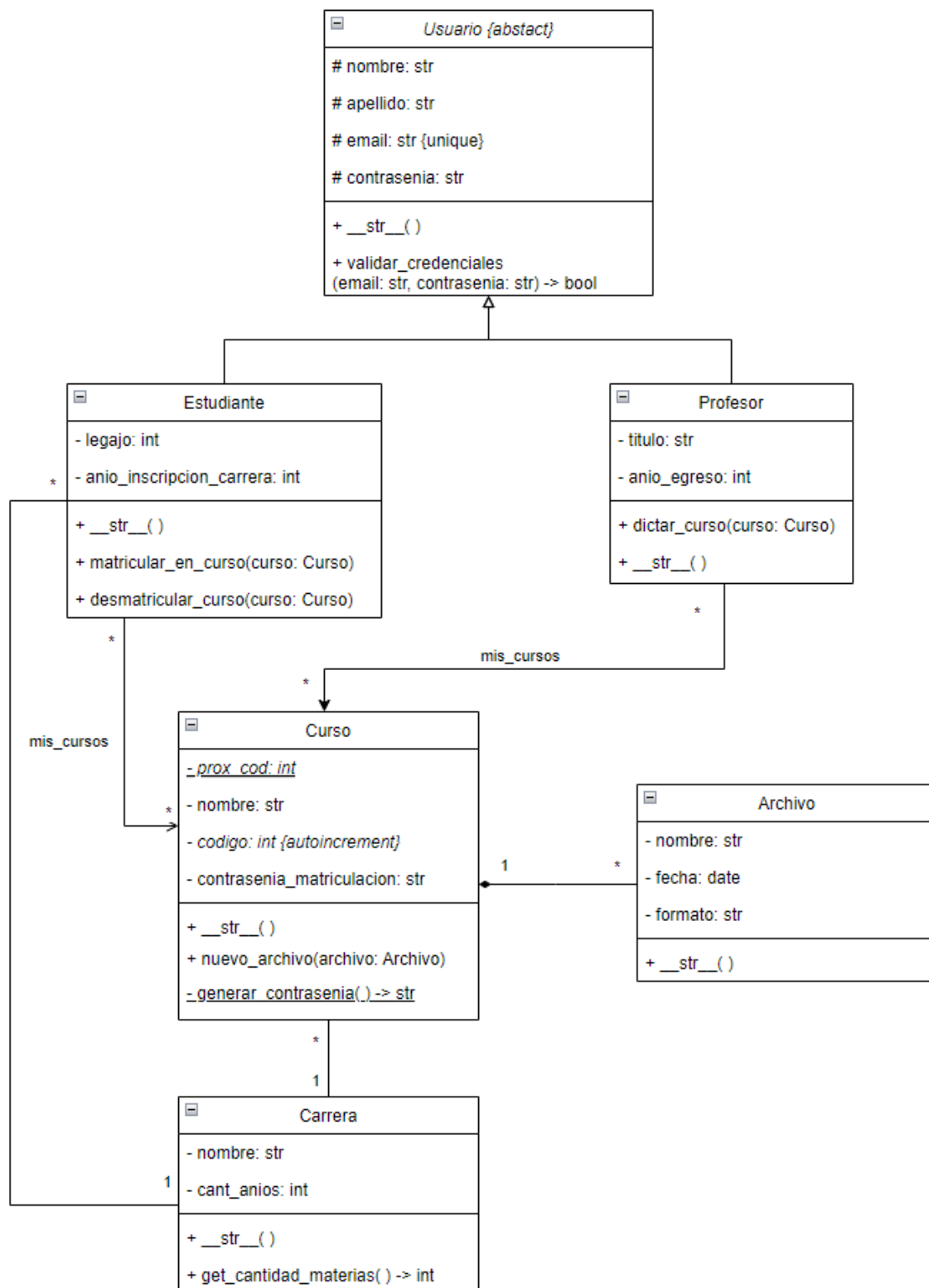
Se deberá entregar en un documento .pdf la siguiente información:

- Título: Trabajo Práctico Integrador
- Entrega: Final
- Comisión: XTUPX
- Integrantes: (nombre - legajo)
- Repositorio: Link al repositorio con el trabajo práctico
- Informe del código analizado.

Además se deberá entregar un .zip con el código.

Parte 1

Extender el código de la primera entrega para que respete el siguiente diagrama UML en Python:



Se debe respetar los siguientes requerimientos

El sistema cuenta con algunos alumnos y profesores dados de alta. También se cuenta con una carrera dada de alta. No se cuentan con cursos ni archivos dados de alta al iniciar el sistema.

Al ingresar al sistema se debe mostrar un menú en bucle con las siguiente opciones:

1. *Ingresar cómo alumno*
2. *Ingresar cómo profesor*
3. *Ver cursos*
4. *Salir*

Cuando se selecciona la opción 1 se ingresa el email y la password. Primero se busca con el email en la lista de alumnos registrados si el mismo existe.

Si existe se valida que el email y contraseña sean correctos, si son correctos se da acceso al sistema. En caso contrario se muestra un cartel de error de ingreso.

Si el alumno no existe con ese correo se debe mostrar un mensaje diciendo que debe darse de alta en el sistema.

Cuando se selecciona la opción 2 se ingresa el email y la password. Primero se busca con el email en la lista de profesores registrados si el mismo existe.

Si existe se valida que el email y contraseña sean correctos, si son correctos se da acceso al sistema. En caso contrario se muestra un cartel de error de ingreso.

Si el profesor no existe puede darse de alta ingresando el código *admin* para acceder al alta del profesor. Luego de cargar los datos el mismo se agrega a los profesores dados de alta y se regresa al menú principal.

Cuando se selecciona la opción 3, se debe mostrar una lista de todos los cursos del Campus Virtual ordenados alfabéticamente como sigue:

<i>Materia: Ingles I</i>	<i>Carrera: Tecnicatura Universitaria en Programación</i>
<i>Materia: Ingles II</i>	<i>Carrera: Tecnicatura Universitaria en Programación</i>
<i>Materia: Laboratorio I</i>	<i>Carrera: Tecnicatura Universitaria en Programación</i>
<i>Materia: Laboratorio II</i>	<i>Carrera: Tecnicatura Universitaria en Programación</i>
<i>Materia: Programación I</i>	<i>Carrera: Tecnicatura Universitaria en Programación</i>
<i>Materia: Programación II</i>	<i>Carrera: Tecnicatura Universitaria en Programación</i>
...	

Sub menú alumno (cuando se ingresa como alumno exitosamente):

1. *Matricularse a un curso*
2. *Desmatricularse a un curso*
3. *Ver curso*
4. *Volver al menú principal*

Cuando se selecciona la opción 1, se debe mostrar una lista de todos los cursos del Campus Virtual que pertenecen a la carrera del alumno ordenados por código como sigue:

- 1 *Programación I*
- 2 *Programación II*
- 3 *Laboratorio II*
- 4 *Ingles I*
- 5 *Ingles II*
- ...

Se valida que se ingrese una opción de las mostradas (1,2,3,4,5,...) y se debe validar que el alumno no se encuentre ya matriculado en dicho curso, es decir que el curso no se encuentra en su lista de mi_cursos. Además se valida que el alumno pertenezca a la carrera a la que dicho curso pertenece y que la contraseña de matriculación ingresada sea la del curso. Si se cumplen todas las validaciones se agrega el curso a los cursos del alumno en la lista mi_cursos. En caso contrario se muestran mensajes correspondientes.

Cuando se selecciona la opción 2, se muestra una lista de los cursos que el alumno se encuentra matriculado cómo sigue...

- 1 Programación I
- 2 Programación II
- 3 Laboratorio II

...

Se valida que se ingrese una opción de las mostradas (1,2,3,...) y se elimina dicho curso de la lista mi_cursos.

Cuando se selecciona la opción 3, se muestra una lista de los cursos que el alumno se encuentra matriculado cómo sigue...

- 1 Programación I
- 2 Programación II
- 3 Laboratorio II

...

Se valida que se ingrese una opción de las mostradas (1,2,3,...) y se muestra el nombre del curso y todos los archivos ordenados del más antiguo al más nuevo, por ejemplo...

- Programación I
- tpi.pdf
- practica1.pdf

Sub menú profesor (cuando se ingresa cómo profesor exitosamente):

- 1. Dictar curso
- 2. Ver curso
- 3. Volver al menú principal

Cuando se selecciona la opción 1, se dará de alta un curso para la carrera ingresada y nombre del curso. La contraseña de matriculación se genera automáticamente. El código del curso se asigna al momento de crear el curso. Al dar de alta un curso, el mismo no tiene adjuntos. Luego de creado el curso el mismo se agrega a la lista mis_cursos del profesor y se deja disponible en el sistema para la matriculación de los alumnos.

Se muestra un mensaje de éxito y todos los datos del curso dado de alta...

- Nombre: Programación I
- Código: 10
- Contraseña: kKj10

Cuando se selecciona la opción 2, se muestra una lista de los cursos que el profesor posee cómo sigue...

- 1 Programación I
- 2 Programación II

...

Se valida que se ingrese una opción de las mostradas (1,2,...) y se muestra el nombre del curso, la contraseña, código y la cantidad de archivos que posee, por ejemplo...

Nombre: Programación I

Código: 10

Contraseña: kKj10

Cantidad de archivos: 5

Además se debe mostrar un mensaje de si desea agregar un archivo adjunto, si el usuario ingresa una respuesta afirmativa el profesor ingresa nombre y formato del archivo, la fecha es la del día, se da de alta dicho archivo y se agrega a la lista de archivos del curso. Al finalizar el alta del archivo se vuelve al submenú del profesor.

Parte 2

Para esta entrega final se les asignará una solución de otro equipo realizada en la primera entrega a la que se le deberá hacer un informe con un análisis del código.

Ejemplo de cuestiones a observar:

- Adecuación del código al Pep8
- Adecuación a los requerimientos indicados
- Interpretación del diagrama UML y su correcta codificación en python
- Estructura del proyecto
- Descripción de errores encontrados, cantidad y severidad.
- Pruebas realizadas, cuales pasaron y cuales fallaron.

Se deberá hacer un refactoring de una porción de código que encuentren mejorable. Pegar el antes y después del refactoring en el informe.



Versiones

Fecha	Versión
02/10/2023	1.0
17/10/2023	1.1 - Cambio visibilidad en parámetros clase abstracta Usuario