|  |  |
| --- | --- |
| **Universidad de Costa Rica**  **Facultad de Ingeniería**  **Escuela de Ciencias de la Computación e informática** |  |

**Curso CI-0123 PIRO**

**Proyecto Integrador**

Entregable #1

**Profesores:**

Francisco Arroyo Mora

José Antonio Brenes Carranza

**Asistente:**

Lester Cordero Murillo

**Estudiantes:**

(B62110) Estudiante1 Apellido1

(B62110) Estudiante2 Apellido2

(B62110) Estudiante3 Apellido3

(B62110) Estudiante4 Apellido4

(B62110) Estudiante5 Apellido5

**II Semestre, 2020**

**1. Introducción**

Utilice esta sección para mencionar aspectos generales acerca del entregable, como por ejemplo: cuál es la funcionalidad esperada, cuál fue el propósito y qué conocimientos relacionados a la materia vista en clase se tomaron en consideración para esta solución.

**2. Objetivo General**

A continuación se establece el objetivo general del proyecto:

* Objetivo 1
* Objetivo 2

**3. Objetivos Específicos**

A continuación se establecen los objetivos específicos para esta etapa del proyecto:

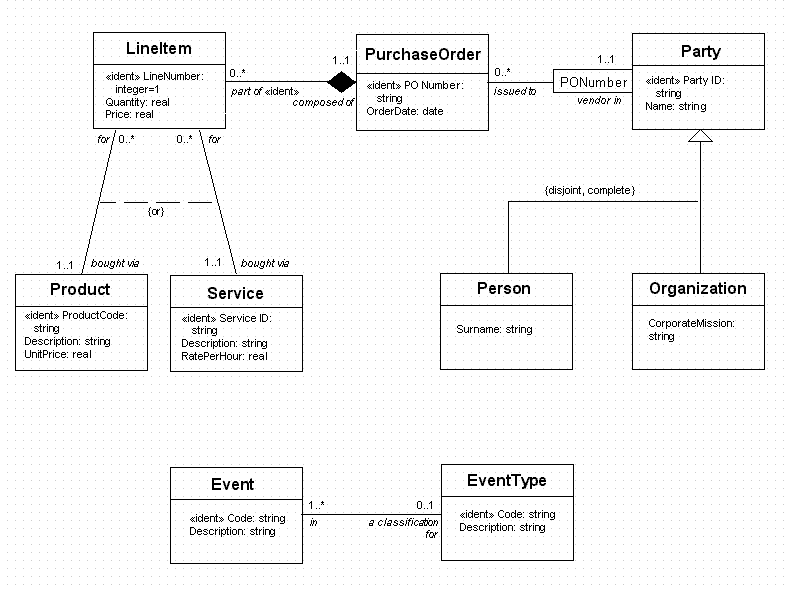
* Objetivo 1
* Objetivo 2

**4. Descripción de la Solución**

Explique cómo se resolvieron los objetivos de esta etapa del proyecto. Especifique las pautas necesarias que se consideraron para lograrlo y qué inconvenientes se presentaron. Coloque aquí las suposiciones.

**5. Esquema de la Solución**

Utilice un esquema de clases UML, en Umbrello, o draw.io, etc o similares.



**6. Arquitectura de la Solución**

Explique brevemente qué hace cada clase. Así mismo, detalle cómo se manejaron los errores o entradas fallidas e incorrectas. Indique si se usaron librerías o frameworks y cualquier aspecto que considere necesario para entender la solución.

|  |  |
| --- | --- |
| Concepto | Descripción |
| Clase1 | Esta clase se encarga de…. |
| Clase2 | Esta clase se encarga de…. |
|  |  |
|  |  |
|  |  |

**7. Limitaciones de su Solución**

Indique las restricciones que fueron agregadas en la solución. Ejemplo, si el programa no funciona en algunos casos, describir cuáles son esos casos. Si el programa no contiene errores entonces no es necesario que incluya esta sección a menos que lo considere conveniente.

**8. Pruebas de funcionalidad**

Describa los casos de prueba para esta etapa del proyecto y sus resultados esperados. Utilice tablas cuando requiera indicar que se necesita ejecutar una instrucción en terminal.

|  |
| --- |
| make ejemplo  output.exe  comando  cd ejemplo/etc/ |

Sería recomendable que coloque un caso de prueba de éxito y un caso de prueba de error. Por ejemplo, demostrar que el programa NO se cae si se ingresa información inválida.

**9. Referencias Bibliográficas**

Puede usar el Formato APA versión 6 (guía de referencia → https://www.um.es/documents/378246/2964900/Normas+APA+Sexta+Edici%C3%B3n.pdf/27f8511d-95b6-4096-8d3e-f8492f61c6dc):

**EJEMPLO:**

Apellido, Nombre. (2020). Nombre del artículo. Recuperado de:

<http://marcoviaweb.com/estrategias-para-descomponer-historias-de-usuario/>

O puede utilizar el formato de IEEE (guía de referencia → https://www.bath.ac.uk/publications/library-guides-to-citing-referencing/attachments/ieee-style-guide.pdf)

**EJEMPLO:**

[1] <http://marcoviaweb.com/estrategias-para-descomponer-historias-de-usuario/>

**NOTA:**

**Para ambos casos se recomienda hacer uso de un gestor de referencias para simplificar la generación de las referencias.**