



**Tecnológico
de Monterrey**

Situación Problema

Héctor Robles Villarreal A01634105

Martín Alejandro Hermosillo García A01634552

Programación Orientada a Objetos

Fabiola Uribe Plata

Grupo 11

Tecnológico de Monterrey Campus Guadalajara

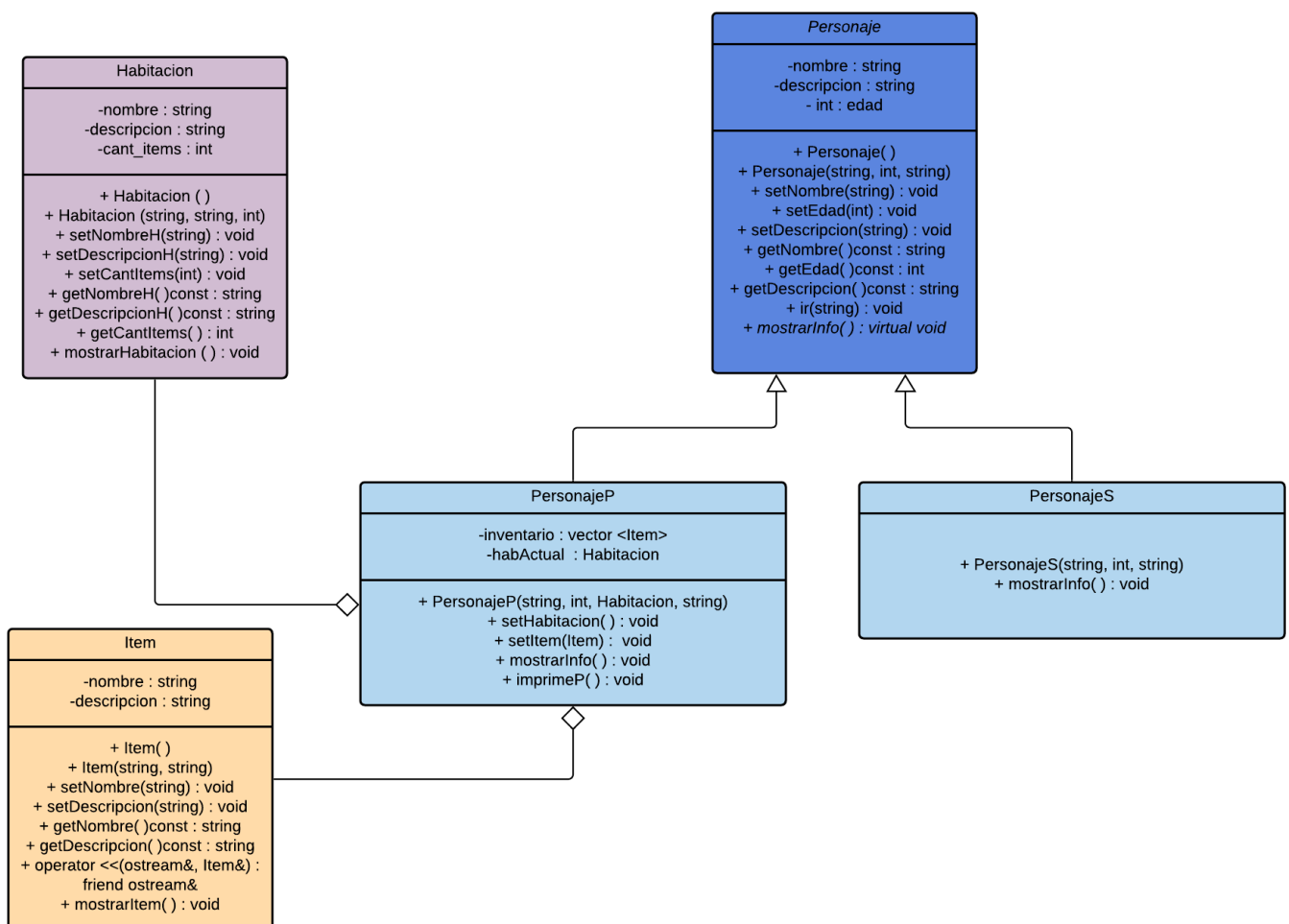
Viernes 11 de junio del 2021

Situación Problema

Como reto de la unidad de formación TC1030 se tuvo que desarrollar un Juego Serio de tipo Aventura Conversacional. Este tipo de videojuegos le presenta al usuario la descripción de la situación por medio de un texto. El jugador posteriormente debe teclear la acción a realizar.

La historia del juego desarrollado, llamado *Das geheimnisvoll Geheimnis* (El Misterio Misterioso en alemán) consiste en tomar el rol de una muchacha que descubre que sus padres están devastados por un trágico suceso que ocurre en un hospital. La muchacha decide entonces indagar acerca de este trágico evento ya que sus padres no le hacen caso. Durante el juego, recorrerá el hospital buscando pistas, las cuales le ayuden a resolver el misterio. Las decisiones del jugador afectan directamente el desarrollo del juego. La historia completa del juego se encuentra en el main del videojuego ya que es de extensión vasta y además presenta partes de historia que no ocurren a menos de que el usuario tome ciertas decisiones.

Diagrama de las clases UML final



Al analizar la situación problema y tener que diseñar las clases que posteriormente serían implementadas, se concluyó que tendría que haber 5 clases totales. Está la clase *Personaje*, que es abstracta y de ella se derivan *PersonajeP* y *PersonajeS*. Es aquí donde se decide implementar herencia. Esta decisión se debe a que las clases con herencia tienen una relación es-un. Al analizar las clases que vimos necesarias, detectamos que tanto el personaje principal (*PersonajeP*), como los personaje secundarios (*PersonajeS*) son un personaje. Estas subclases son una especialización de la clase base y esto nos permitió utilizar métodos de acceso, modificadores y métodos creados por nosotros mismos en estas clases derivadas. Al ya tener herencia entre estas clases, se decidió implementar un método que mostrara la información del personaje utilizando polimorfismo. Este método era virtual en la clase *Personaje*, pero al sobrescribirlo en las subclases, nos permite desplegar la información de este tipo de objetos de distintas maneras. Es ahí claro el uso de polimorfismo en tiempo de ejecución a través de funciones virtuales. Aparte de estas, se cuenta con las clases *Habitación* y *Item*, que tienen una agregación con la clase *PersonajeP*. Para facilitar el despliegue de objetos *Item* se sobrecargó el operador de salida (<<) para hacer más práctico el programa. Finalmente, al haber tenido que utilizar muchos ciclos condicionales para que el desarrollo del juego se modificara basado en las decisiones del usuario, se aplicó una excepción al momento de recibir el input del usuario de la decisión. Si este input no es un valor que el programa pueda leer, se le presenta un mensaje al usuario que le notifica esto.

El proyecto no dejaría de funcionar bajo ninguna circunstancia, esto es debido a que, como mencionamos con anterioridad, se aplicaron los conocimientos adquiridos sobre manejo de excepciones y nos encargamos de asegurar la fluidez y la continuidad de nuestro programa. Lo único que podría causar algo raro, sería el introducir un espacio en la terminal ya que se pueden introducir los espacios que sean, sin embargo, aunque el usuario ingrese muchos espacios y acto seguido teclee algo, el programa detectará esa entrada y actuará en consecuencia, si es un valor válido seguirá con la historia y en caso contrario, le especificará al usuario que lo que ingresó es inválido y que por favor ingrese algo válido.

Conclusiones

Héctor : Considero que este proyecto fue de gran ayuda para terminar de llevar los conceptos teóricos aprendidos en las diferentes clases a la práctica. Esta situación problema engloba de manera puntual todo el contenido visto a lo largo de la materia y te permite poner en práctica las habilidades y competencias adquiridas durante el curso. Personalmente, disfruté bastante realizar este proyecto. El hecho de programar un videojuego me emocionó mucho. Estoy completamente satisfecho con nuestro desempeño y el producto final.

Hermosillo : Me voy muy satisfecho con esta situación problema. El que haya sido un videojuego es algo que despierta mi interés y por ende no se sentía como trabajo, sino como un proyecto personal. Fue una manera excelente para reforzar los conceptos adquiridos en clase y darles una aplicación real. Al momento de intentar resolver el reto, si se presentaron algunas dificultades. Estas más que nada eran encontrar la manera más práctica y sencilla de realizar nuestra visión del lo que queríamos en el proyecto. Afortunadamente, estas

dificultades son las que vuelven interesante al reto, porque nos obligan a pensar e intentar buscar una solución.

Referencias

“C++ Library - <vector>”. (s.f). *Tutorialspoint*. Recuperado de

https://www.tutorialspoint.com/cpp_standard_library/vector.htm

Educación 3.0 (2018) Serious games: los videojuegos también educan. Consultado

en:<https://www.educaciontrespuntocero.com/noticias/serious-games-frenar-acoso-escolar-fomentar-la-creatividad/>

Gamelearn (2017) Todo lo que necesitas saber sobre los serious games y el game-based

learning, explicado con ejemplos. Consultado

en:<https://www.game-learn.com/lo-que-necesitas-saber-serious-games-game-based-learning-ejemplos/>

Observatorio de Innovación educativa, ITESM (2018) ¿Qué son los serious games?

Consultado en:<https://observatorio.tec.mx/edu-news/que-son-los-serious-games>