

Universidad de Sevilla

Escuela Técnica Superior de Ingeniería Informática


DP2-Informe de Análisis Student 4 D2



Grado en Ingeniería Informática – Ingeniería del Software

Diseño y Pruebas II

Curso 2024 – 2025

<u>Group:</u>	C1.018
<u>Repository:</u>	https://github.com/alenicbra/acme-ans
<u>Student #4</u>	
	
UVUS: josmirmar2	
Name: José Manuel Miret Martín	
Email: josmirmar2@alum.us.es	
<u>Date:</u>	Sevilla Mayo 26, 2024

Índice de contenido

1. Versiones	2
2. Lista de los registros de análisis	2
3. Conclusión del análisis	2
4. Enlace de la validación del profesor	2
5. Bibliografía	2

1. Versiones

Versión	Fecha	Autor
1.0	26/05/2025	José Manuel Miret

2. Lista de los registros de análisis

-Task 3: The assistance agents are the people responsible for recording and managing post-flight incidents reported by passengers. The system must store the following data about them: an employee code (unique, pattern "[A-Z]{2-3}\d{6}\$", where the first two or three letters correspond to their initials), a list of spoken languages (no longer than 255 characters), the airline for which they work, the moment on which they began to work for that airline (in the past), and optionally, a brief bio (up to 255 characters), their salary, and a link to a photo that should be stored anywhere else.

Task 4: A claim is a formal request or complain made by a passenger or customer due to a problem or inconvenience experienced during a flight. They are registered by the assistance agents, and the data to store when registering a claim is the following: the registration moment (in the past), the passenger email, a description (up to 255 characters), a type ("FLIGHT ISSUES", "LUGGAGE ISSUES", "SECURITY INCIDENT", "OTHER ISSUES") and an indicator of whether the claim is accepted or not.

Task 5: Claims need to be tracked through tracking logs. A tracking log records each step in the procedure followed to resolve or reject a claim, ensuring that all actions and decisions are documented. The system must store the following data about tracking logs: the last update moment, the step undergoing (up to 50 characters), a resolution percentage, and an indicator on whether the claim was finally accepted or not. When a claim is accepted or rejected, the system must store its resolution indicating the reason why was rejected or the compensation to offer (up to 255 characters).

Task 6: Produce assorted sample data to test your application informally. The data must include two assistance agent accounts with credentials "agent1/agent1" and "agent2/agent2". Create an additional agent account with credentials "manager3/manager3" that accounts for a new agent with no associated data, except for his or her profile.

Task 7: Provide a link to your planning dashboard in GitHub to review the tasks, their current status, and your schedule.

-Task 15: The system must handle assistance agent dashboards with the following indicators:

- The ratio of claims that have been resolved successfully.
- The ratio of claims that have been rejected.
- The top three months with the highest number of claims.
- The average, minimum, maximum, and standard deviation of the number of logs their claims have.
- The average, minimum, maximum, and standard deviation of the number of claims they assisted during the last month

Task 16: Produce a UML domain model regarding the information requirements in your project.

- Task 27: Produce an analysis report.

- Task 28: Produce a planning and progress report.

3. Conclusión del análisis

- Task 4: Con esta tarea hice de analista para entender el requisito de información y de desarrollador para hacer el código. Fue una de las tareas que más me costó elaborar, no tanto por su complejidad, sino porque fue la primera que realicé. Afortunadamente, recibí ayuda de mis compañeros para completarla. Inicialmente, desarrollé la entidad Claim con todas sus propiedades y anotaciones correspondientes. Para manejar las propiedades 'IndicatorType' y 'ClaimType', creé dos ENUM. Además, tuve que establecer una relación 'ManyToOne' con la entidad 'AssistanceAgent' y 'Leg' como parte de los requisitos de información. A través de este proceso, adquirí una comprensión más profunda de la estructura del proyecto y de cómo manejar las relaciones entre entidades.

- Task 5: Para esta tarea tuve que hacer de analista para estudiar lo que debía hacer, y de desarrollador para implementarlo. Esta tarea también fue costosa de elaborar, debido al anterior requisito de información. Aunque ya haya tenido experiencia de la anterior tarea, tuve que elaborar algunas modificaciones en base a los comentarios del professor en clase de practices. Inicialmente, desarrollé la entidad 'TrackingLog' con todas sus propiedades y anotaciones correspondientes. Tuve que establecer una relación 'ManyToOne' con la entidad 'Claim' como parte de los requisitos de información.

- Task 3: Como en las 2 anteriores, llevé a cabo el rol de desarrollador para implementar y de analista para examinar qué tengo que codificar. Esta tarea me llevó menos tiempo en comparación con las dos anteriores. Sin embargo, invertí más tiempo del esperado en examinar los requisitos de información asociados a esta tarea. Afortunadamente, con la ayuda de mis compañeros, pude elaborar la entidad correctamente y sin ningún problema ni inconveniente. En esta tarea tuve que crear el Realm 'AssistanceAgent'. No fue tan costoso de hacer, a excepción de la validación personalizada para crear la propiedad employeeCode.

- Task 6: Para esta tarea llevé a cabo el rol de desarrollador. Tuve algunos problemas a la hora de elaborar los ficheros en formato CSV. Dicho problema era principalmente sobre las columnas de las mismas, ya que no se mostraban de forma adecuada. A su vez me llevó un poco de tiempo comprender cómo se debían de establecer las relaciones entre las entidades. Además, la elaboración de dichos CSV me hizo realizar algunas modificaciones en las entidades. A parte de

dichas cuestiones, no hubo más inconvenientes.

- Task 15: Para realizar esta tarea tuve que desempeñar 2 roles. El primero fue el de analista, ya que tuve que estudiar las restricciones y propiedades dadas en el requisito de información, y a partir de ello elaborar el UML de las entidades. Con el segundo rol (desarrollador) llevé a cabo el código correspondiente. El análisis de la entidad fue bastante sencillo debido a mi experiencia previa en las anteriores tareas. A su vez, las restricciones eran bastante sencillas tratándose de simples restricciones para cada propiedad.

- Task 16: Realizar el UML fue sencillo, debido a que obtuve ayuda por parte de mis compañeros. Desempeñé el rol de Analista para esta tarea. La única complicación que tuve fue la propiedad 'duration', comentada anteriormente. A parte de eso no hay mucho más que comentar sobre esta tarea.

- Tarea 27: En esta tarea asumí el rol de project manager. No encontré dificultades significativas ya que simplemente tuve que plasmar las decisiones de diseño y los roles que desempeñe para las demás tareas. Además, la estructura ya estaba definida a partir del entregable anterior, lo que facilitó el proceso. En resumen, no hubo problemas destacables y no hay mucho más que añadir al respecto.

- Tarea 28: En esta tarea también actué como project manager. No fue complicado analizarla ya que al leerla ya había revisado las tareas anteriores. Esto me permitió tener una idea clara de cómo organizar todo, y gracias a la experiencia adquirida en la entrega anterior, estructurar el documento resultó más sencillo. No fue necesario tomar decisiones adicionales de diseño, dado que la estructura del documento ya estaba establecida previamente.

4. Enlace de la validación del profesor

En esta entrega no hay enlace a validaciones del profesor.

5. Bibliografía

No hay bibliografía presente para esta entrega.