Digital business model

Final task one

Submitted by

Denil Tony

Introduction

Now days, e-commerce industry is growing very fast, it means the customer interactions, sales and transactions are highly dependent on correct and well-structured software system. Such systems are complex, and hence need a strong method to support designing, visualizing, and communicating the structure. Meanwhile the tool or rather the language that the developers and system architects use to model the structure and behavior of these complex systems is Unified Modeling Language (UML). There is article that describes how this method helps the teams which develop system applications by providing all necessary visuals to create an effective system that meets all the necessary business needs and requirements of its users.

This paper explores how UML has been applied in the development of five different e-commerce platforms: Its outlets include AudioTech Electronics Store, Vivo House Showroom, Cilla Clothing Store, FGH Hypermarkets and a general E-business selling platform. All of these platforms employ Use case Diagrams, Activity Diagrams, Sequence diagrams and Class diagrams to model the system architecture of the applications. In this paper, we will look at ways that they relate to system clarity, communicating, development and hence customer experience and organizational efficiency.

- **Visual Representation**

The first benefit of using UML in e-commerce is that it offers customers **models** of the computer system and its elements. Class and sequence diagrams are used to show the association between and among different entities, for instance, customers, the products, and the orders. This feature enables the team and other users of a system design to have a quick and easy way of making conclusions and comprehending the construction of the system. For example, in the form of UML diagrams, one can perfectly describe the structure of an e-shops and describe such classes as `Customer`, `Product`, `ShoppingCart` and `Order` as well as their attributes and methods.

- **Standardised Language**

UML is formally defined as a **standardised and comprehensive language for capturing information about software intensive systems. Through the presence of one established notation, the developer can easily explain their concepts to business analysts, project managers and other stakeholders as well. This mutual understanding eliminates any confusion and sets an expectation which can be helpful to manage multiple stakeholder groups present in e-commerce.

- **Model-Driven Development**

There is another important advantage that belongs to UML – **model-driven development**. Developers can model in details laying grounds for the structure of the system. These models can then be pushed straight into code or some other artifact as part of development. Through the code generation from the UML diagrams, much effort is exercised towards reducing common mistakes likely to occur during the actual coding process hence improving on the rate of work.

- **Collaboration**

Unified Modelling Language encourages the working of diverse **teams** in the development of a software system. Miro, for instance, provides templates for UML diagrams that will enable the team to collaborate in real-time that allows for teamwork in terms of brainstorming and giving feedback. This makes it not only foster creativity but also make sure all design possibilities are covered before implementation. For instance, a sequence model using UML may be employed to define the encounter map when checking out the products and used to notice any future issues inside the implementation stage.

Unified Modelling Language Role in System Design

UML is a structured form of specifying large software systems that relies on graphical models. In the environments of e-commerce platforms, the interactions with thousands of customers, products, and transactions occur often, UML serves as a necessary means to organize the layout of the system's static structure and its dynamic behaviour in mind.

UML is crucial in carrying out the design of the AudioTech e-commerce system. By describing the structure and behaviour of the system with the help of various types of UML diagrams, the developers were able to see how this or that part of the system would interact with other parts of the system. This allowed for the creation of a simple and easy to navigate website that prospective customers were able to create accounts, search for the audio products of their preference, make an order and even check on the progress of their order. Since the work was done with the help of UML, all the parts of the system, starting with the customer interface and ending with the back end such as inventory storage, were described and well documented.

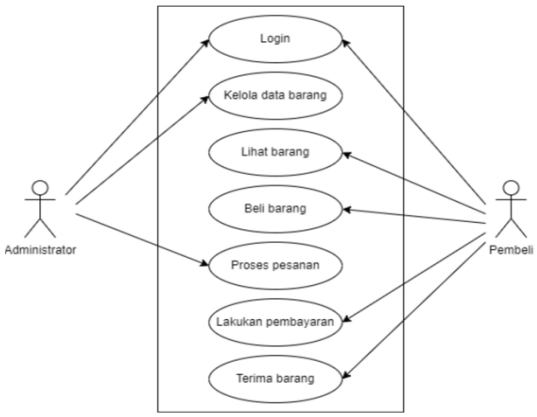
UML was used in the development of the Vivo Store which aimed at ensuring that a mobile phone sales platform was designed in an efficient manner in a way that would easily be understood by the customers and the administrators. The diagrams assisted in mapping out the customer process of surfing through available products, making orders, and making payments among others, and the administrative process which included putting into consideration the stock of products and sorting out customer orders among others. Thus, with the help of UML, developers were able to make a provisionage that the given system will correspond to the necessities of the business from the viewpoint of functional-tasks and, at the same time, will possess the characteristics of scalability and relative modifiability.

The Cilla Shop and FGH Stores systems applied UML in the same manner as an e-commerce platform to model their systems. These systems necessary to reorganise customer interactions and many other administrative procedures. UML helped the developers to conceptualise platforms that provided effective user interfaces and generative business environments in the back end.

In summary, UML is involved in all of these systems simply because it gives the developers a map to follow. This makes it easier to comprehend, create and manage such complicated e-commerce solutions resulting into enhanced effectiveness, productivity and consequently customer satisfaction.

Use Case Diagrams in E-Commerce Systems

Use Case Diagrams is one of those UML diagrams that are the most popular in system designing, especially in e-commerce. They depict how the users perform tasks with the intended system, and clearly describe the overall major functionalities, and orient the developers of the intended system on how it must be constructed to suit the users.



(*audio-tech use case diagram)*

The Use Case Diagram for the AudioTech e-commerce system shows the Customer Use Cases for the e-commerce system. Specific scenarios deal with client identification and authentication, navigation and searching for products, placing items in a shopping cart, and purchase of the items. These diagrams help the developers give them a picture of how the system will work and how the customer will be able to use it. These were core functionalities that, if identified by the development team, they could be able to ensure that the system was friendly, optimum, and customizable.

Another example of the software's ability to use Use Case Diagrams can be seen in the case of the Vivo Store e-commerce platform. On the side of the customers, the following are shown as a flow chart: logging into the site, viewing products, purchasing products, and completing the payment. For administrators, effective action options are presented within the diagram,

including the options of viewing the stock of phones, managing orders, and validating transactions. These use cases define the functional flow of the system and also make sure that both parts of the customer, as well as administration, are adequately covered.

In Cilla Shop, the Use Case Diagram focuses on two key actors: the customer and the admin. The customer engages with the system through product navigation, ordering, and check orders and products. While the admin deals with category, products, customers, and their transactions. There exists a lot of difference between one user type and the other, which is paramount when attempting to understand how each one of them or both will interact with the actual system. Such segregation of features is evident from the diagram where the customer-oriented aspects involve convenience to gain access to the product in question, whereas the admin-oriented aspects address the problem of managing data and transactions.

Likewise, in FGH Stores system, Use Case Diagramming is employed to focus on the various ways that the customers and the administrators may interact. For customers, navigation options include viewing products, creating accounts and ordering, as well as monitoring order deliveries. To illustrate, the diagram identifies activities such as product information management, order processing, inventory tracking, and sales reporting for administrators. Developing these use cases and then modelling them visually allowed the developers to construct a system that was not only successful at handling customer transactions but also at maintaining administrative organization. By the use of this diagram, it is possible to avoid the creation of a system that is built on ambiguous functional requirements that may not meet the user needs.

The Use Case Diagram also has its application in the generic e-commerce sales system where customers are involved in browsing products, ordering products among other activities. In the same regard, administrators must control the inventory, orders and customer's details as well. This diagram ensures that different developers and other stakeholders that are involved in the development of the e-commerce system are on the same page in terms of the functionalities of the proposed system and other features that comprises of product catalog and payments.

Use Case Diagrams play important roles in converting business specifications into working system models. Thus, describing the user interactions, they present the developers with a plan which they can follow to deliver the functionalities that will be useful to the customers as well as the administrators. This structured approach assists in ensuring that the system stays fully focused on the user's experience with operations in the background being done effectively.

Activity Diagrams for Workflow Visualisation

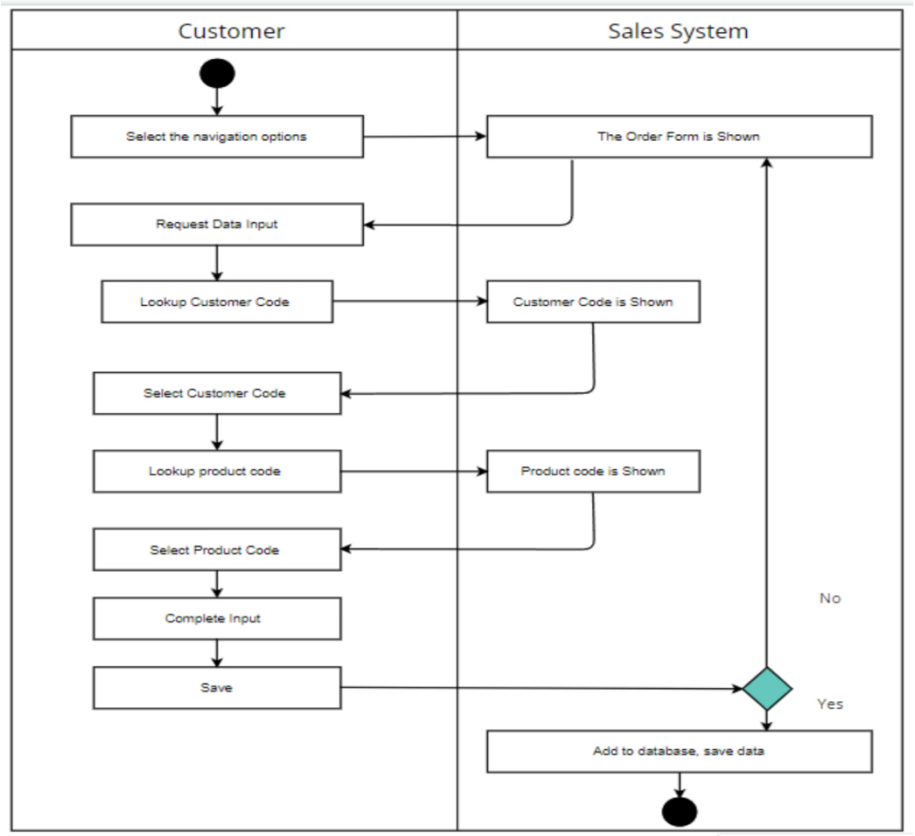
Activity Diagrams are used to compliment the use case maps in that they offer a more detailed look at the activities that goes on in the system. Organisations such as the software development company prefer flow diagrams because of the following reasons: They makes each process to

look like a sequence of isolated steps, thus facilitating easy comprehension of how different activities are performed and guaranteeing examination of all doable steps.

In the AudioTech e-commerce system, the Activity Diagram is employed to model such as user registration and order placement workflows. For instance, in the registration process the user completes the form and it is followed up by validation for the preformed form by the system. If the form is filled, then the registration is complete, and the user is given the access to their account. Further, the order placement work flow describes the ability of the customers to be able to choose the product they want and pay for it through entering various details to make the purchase. These diagrams make sure that all the process is well elaborated, and that the system can respond to each action without failure.

To Vivo Store, Activity Diagram provides ways of modelling customers` and administrators` activities. Customers' actions such as entering their account, viewing goods, ordering, and canceling an order are shown in a linear way. The complete work flow in the admin section including the order management, deleting the product and keeping the track of order placements are described to make organised each work flow. These diagrams emphasise some paths of a work flow and allow to determine, for instance, which path affects the system's performance most, or, in other words, which path can be a critical one.

At Cilla Shop, activity diagrams describe both external processes, which are visible to customers, and internal processes, such as clerical work. For instance, the customer workflow might involve activities such as, browsing through the store; putting items in the cart; and going through the order checkout process. While on the administrative level, these consist of bringing new products into the system, handling customer's orders and preparing sales reports. But by visualising these workflows the developers can make sure each process is as improved as possible and that none of the potential problems – like failed payments or missing products – are being overlooked throughout the development phase.

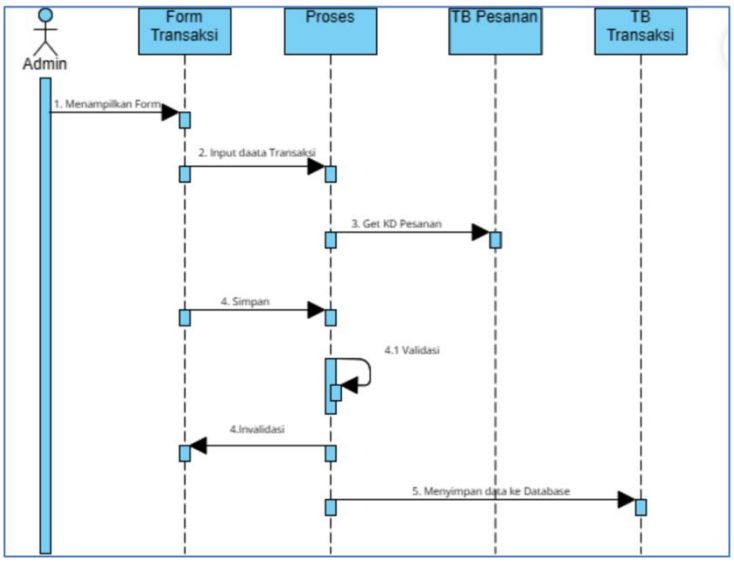


(FGH active diagram)

Likewise, at FGH Stores, Activity Diagrams are employed to model work flows including customer registration/ order management. Herein, the strategies of registration contain using a form, input validation, and then confirmation. The order management is the process with numerous actions including the body actions such as check for product availability, payment actions as well as action of updating the inventory. Through such workflows, the developers manage to design the systems in such a way that all functions run as intended and all operations are optimized.

In all these cases, Activity Diagrams become useful, facilitating a depiction of the many details concerning the operations within the system. They assist the developers in figuring out the correct flow and procedure and guarantee that every point in the process is explained. These diagrams also make it easier for developers to understand where the process can be refined and where bottlenecks can develop, if one is looking at the process in terms of the amounts of time various steps require.

Sequence Diagrams for Interaction Modelling



(sequence diagram)

Sequence Diagrams are used in descriptions of patterns of communication over the time between objects or components of the system. These diagrams are most useful in knowing more about the behaviours of the system depending on user inputs as well as coordination of various components in executing a particular function.

At the e-commerce system of AudioTech the Sequence Diagram represents the possible interactions of the customers and the system during the ordering of products. Example, when a customer chooses an order, the system also examines to identify whether the product is in stock or not, then examine the payment and then moves on to updating the stock. Each interaction is depicted here as a message passed from one object to another so that developers can clearly follow a series of actions performed. This is important so as to allow the system to meet the needs of the users by responding to the input from these users in realtime and also to ensure that every process that is in the system or part of it is done perfectly the first time.

The above figure also shows that the Vivo Store e-commerce platform also employs Sequence Diagrams to show the kind of interactions expected from customers and the system. For instance, the diagram could depict a buyer's process through which a particular person navigates through products, selects the desired product, and makes payments. When a customer goes through the catalog, through the shopping cart up to the payment gateway, all the relations depicted in the diagram become clearer to the developers. Same to this, the interaction between the admin and the system is shown; such as processing the product update request, inventory check request, order validation request, etc.

In Cilla Shop, Sequence Diagrams show the interactions between the customers, the admin, and the system during some significant activities such as order delivery and customer services. For instance, the diagram may illustrate how the contacts are made each time the client places an order; stock check, order confirmation, and updating the transaction log. By citing such

interactions, the developers will be in a position to know how the system will attend to each of the requests and whether some steps have been omitted.

In FGH Stores, Sequence Diagrams provide a clear view of the interactions during processes like order placement, payment validation, and inventory updates. For example, when a customer places an order, the diagram shows how the system first checks product availability, processes the payment, and updates inventory levels in real-time. This sequence of interactions ensures that no steps are missed and that customer orders are handled in an efficient and timely manner. By visualising how different components interact—such as the customer interface, payment gateway, and inventory system—developers can identify potential bottlenecks or inefficiencies and refine the system accordingly.

In all the systems analysed, Sequence Diagrams serve as a powerful tool for understanding the flow of information and requests between different system components over time. These diagrams are especially useful for identifying potential delays, errors, or redundancies in the system's workflow. For instance, if a certain process takes too long to complete or fails under certain conditions, Sequence Diagrams help developers pinpoint exactly where the issue occurs, allowing for quicker troubleshooting and optimisation.

Moreover, Sequence Diagrams are instrumental in the testing phase of system development. By visually mapping out how the system should behave in response to different user inputs, these diagrams provide a reference point for developers to check whether the system is functioning as expected. This not only improves the overall quality of the system but also helps ensure that the final product is reliable and performs well under real-world conditions.

Class Diagrams for System Structure and Database Management

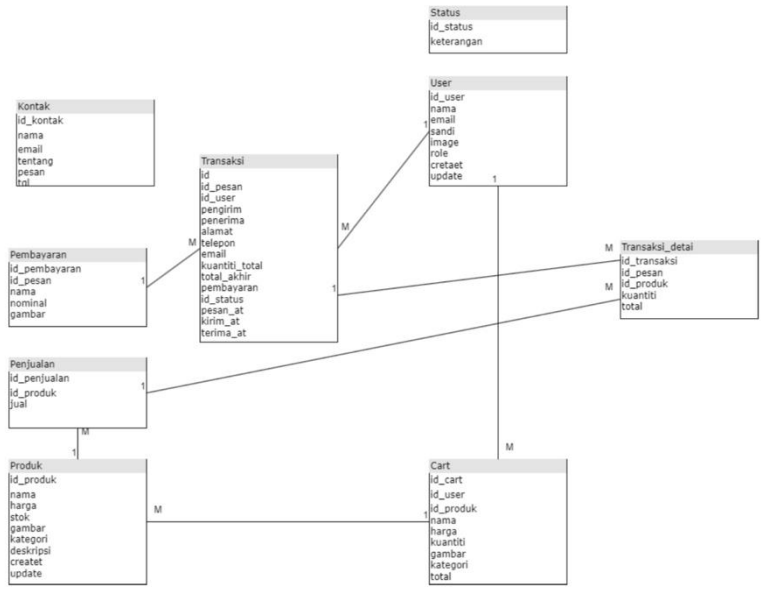
Class diagrams are used in OOPS and DBMS design. These diagrams are either static as they present the relation between classes (entities) in the identified system. This often comes in handy to answer questions relating to how data in the system is organized and how the various elements including but not limited to the products, customers or transactions relate to each other.

When it comes to AudioTech e-commerce system, the Class Diagram illustrates data association that connects the crucial entities as: Product, Customer and Transaction. Each class is described with the set of attributes and methods that this class has in this or that role in the system. For example, the Product class may have properties including name of product, its price and description while the Customer class may include properties such as customer ID, customer name and records of purchase. Moreso, the attributions of these classes for example how the Customer class is related to the Transaction class where a customer can have many Transactions and a Transaction can only have one Customer, and the other relationship between

classes such as the Product class to Transaction class depicted in the diagram above are very well defined. It assists the developers in designing the database of the system and guarantee that every piece of data required is collected.

Similarly, in the Vivo Store platform, the static structure of the system is shown by using the Class Diagram where entities like Product, Customer, Order, Payment interact. For instance, the Order class connects it with the Customer and Product classes because each order is made by a customer and may contain one or more products. The Payment class is associated to the Order class, which models the monetary exchange between a customer and a seller or between two sellers. The Class Diagram just described thus provides an outline of how the system's database should be structured and offers a guide on how data should be processed to achieve maximum efficiency.

Auliana, et.al., *ARRUS Journal of Engineering and Technology*, Vol. 4, No. 1 (2024)
https://doi.org/10.35877/jetech2710
OPEN ACCESS
ISSN : 2776-7914 (Print) / 2807-3045 (Online)



(*Vivo class diagram)*

In Cilla Shop too the Class Diagram presents the same entity relationships such as Product, Customer, and Transaction. The same way each class is given certain attributes for instance customer ID, product ID, transaction amount and methods that the system is supposed to execute. This diagram assists in verifying that the system's database is properly constructed and that all needed data is determined and classified in a format conducive to optimum system function.

Thus, for FGH Stores, Class Diagram provides understanding of the many to many relationships between Customer, Order and Inventory. The diagram also captures how each customer can order several products while each order made is associated with both a product and an inventory record. By visualising such relations, developers can create the system

database and thus optimize the inventory and orders tracking. This is especially important in e-commerce systems since real time updating of stock and exact order management play key roles to the delivery of an excellent consumer service.

In total, there is nothing more important for describing the static structures and relations of the e-commerce system and the connections between them than Class Diagrams. These diagrams are helpful because when the developers who are working on the system are designing the system's database, these diagrams give a clear framework, then the data is organized in a manner that will provide efficient operating of the system. Coherent relationships between other entities are also defined clearly in Class Diagrams which would also make one guarantee the ability of the system to expand, or adapt to incoming changes in the future due to expanding business domains.

Importance of UML in E-Commerce Development

The incorporation of UML in developing e-commerce systems has many advantages including enhanced clarity of generated models, enhanced organization and logical practice, and effectiveness in communication because it is an international standard. As demonstrated in AudioTech, Vivo Store, Cilla Shop, FGH Stores, and the generic e-commerce sales system, UML was especially crucial in guaranteeing that all the systems were designed to fit both functionalities and flexibility.

Another benefit of using UML is that it reduces some extent of confusion in designing the future system. It means that by using Use Case Diagrams, Activity Diagrams, Sequence Diagrams and Class Diagrams developers can divide complicated process into weak sign. First, it enhances one's ability to comprehend how the system is supposed to work and second, it allows one to see any problems with the system early enough in development. For example, a Use Case Diagram may show that there are some features that are not included in the system, a Sequence Diagram may show that the flow of the systems may be suboptimal.

UML also aids in structural development in that it outlines a clear plan of the coding and implementation step. For example, with use of Activity Diagrams, the developers are able to capture the logical flow of operations including steps that will be followed in the execution processes, and thereby the operations to be performed is defined comprehensively. As with the above case, Class Diagrams offer a protean that depicts the system's database and consequently the manner in which data would be arranged to facilitate function of the system. The following structural approach aids the developers and guarantees that all aspects of the system are developed in an efficient structurally conceived sequence.

The other essential benefit that comes with the use of UML is that the tool is useful in improving the communication process in the project among the various working teams. Globally, the system ascertains that UML diagrams facilitate an easy coherent approach to describing the functionality of a system in a coherent way thus attain the consensus of the developers, the

stakeholders and the project managers regarding how the system should function. This is important so that all stakeholders in the project understand the requirements and the system implemented suits the project's descriptions. What is more, through the provided notations, UML minimizes the possibility of such a mistake, as well as makes the project less susceptible to misinterpretations.

Besides, UML assistance makes it possible to scale the system and be prepared for future changes. For instance, this is in a manner where, through the use of Class Diagraming, developers are able to identify relations among various entities, of a system's database architecture in the future development of the required solution. In the same way, Activity Diagrams and Sequence Diagrams will assist in discovering the envisioned bottlenecks of the system's flow and modify the development process for amplification. This helps to ensure that a system which will be able to expand in size, or handle more traffic, or other new additional activities or functions without much effort and adjustment.

Challenges and Future Directions

Despite all the advantages that can be provided by the usage of UML, there are some difficulties which can be originated by the developers when employing it for the design of the real complex system. One of the critical issues is concerned with the nature of the models that are usually developed for use with "big data." For complex extensive systems it is possible to end up with numerous and very detailed diagrams in UML. This annoys the developers because it becomes difficult to monitor all the components in the system to make sure that the system is being developed based on the proposed design. One to address this challenge is that developers should try to reduce complexity of the diagrams drawn by fractalicious? Method one is dissecting diagrams to even smaller bits that are easy to manage or, the use of tools that provide for the automatic creation of UML diagrams.

Another issue that we have already mentioned is the adoption cost and the initial adaptation to the use of UML. To the developers who are not conversant with UML, sometimes it will take time to learn how best to use the different types of diagrams. In case of this challenge, the teams are trained on how to go about with best practices on UML and use resources that enable easy understanding of how to create UML diagrams.

In the future, the utilization of the UML in e- commerce system development may change additional to newer technologies. For instance, incorporation of Artificial Intelligence (AI), and machine learning (ML) into e-commerce platforms is evident nowadays. However, as these technologies make their way into the enterprise domain, UML should be able to extend the modelling paradigms to accommodate AI business processes and data-centric transformations. Furthermore, there is more emphasis on the use of 'predictive analytics and the transformation of 'big data' in the field of e-Commerce and intrinsically, UML will have to expand to incorporate the modelling of these types of data handling intensive systems.

Conclusion

The UML is essential in the architecture of e-commerce systems because it offers an organised and graphical way of working on the systems needed for e-business. In any of the e-business sites discussed here: AudioTech, Vivo Store, Cilla Shop, FGH Stores, and the generic e-business sales system, UML significantly assisted developers in modelling how users' interface with the system, how the multidimensional process flows in real-time, and how the databases are structured. Also, in a way best fitting the need, through the Use Case Diagrams, Activity Diagrams, Sequence Diagrams, and Class Diagrams this system were developed and explained in a clear manner with the best scalable output and to boot cut down on the time needed to develop the systems and improve communication between development teams.

With the help of breaking up multi-stage processes, UML facilitates constructing work utility that is simultaneous to being user-friendly and compatible with back-end requirements. The use of the UML also aids in the overall structured development to meet business requirements while also serving to communicate effectively the proposed system to the various stakeholders and give the user an enhanced positive experience while using the system. Thus, UML will continue to be an effective instrument in designing and realizing e-commerce systems with reference to the new emerging technologies, aimed to create effective and prosperous systems, which will provide the necessary features of scalability, flexibility, convenience and efficiency for customers and business companies.

References

1. The Role of E-Commerce in Increasing Sales Using Unified Modelling Language, Djarot Hindarto
2. Information System Design at FGH Stores with Unified Modelling Language Djarot Hindarto, Mochamad Hariadi
3. Designing E-Commerce Applications for Mobile-Based Audioo Technics Stores Muhammad Fajar Fikri, Nurul Hamdi
4. Aplikasi Toko Online Berbasis Web Studi Kasus pada Cilla Shop Pitri, M. Syafiuddin Usman
5. Design of a Mobile Phone Sales System Website at Vivo Store Sigit Auliana , Gelard Untirtha Pratama, Basuki Rakhim Setya Permana*, Sunardi, & Oki Firmansyah

Scenario: Netflix subscription platform

A visitor comes to the Netflix web site or to the Netflix application on a mobile device and scrolls down the list of offers. They can watch some trailers, read descriptions as well as sort films according to the type, for example, action, comedy or documentary and others. However, they will have to subscribe to any stream they desire, if they are to watch any stream at all.

The visitor subscribes by entering the first name and email, then an individual password, and lastly selecting one of the subscription plans: Basic, Standard or Premium. While using the Internet or social networks, or after registering to access higher level content after providing credit card or PayPal details, the person becomes a subscriber.

They were a subscriber and now they have the ability to watch anything that Netflix has to offer. They can search for the available shows or movies and mark them in the liste, for viewing them and can also watch the available programs on the mentioned devices such as smartphones, tablet, smart-tv, and computers etc. It uses a recommendation engine where Netflix proposed materials the subscriber seems to be interested in because of what he has been viewing recently.

Clients have the rights to modify, the type of subscription plan they are using, alter payment methodology or revoke the service provided at any time. That includes the opportunity to create several profiles within one account to provide individual recommendations and watched TV shows for different users– family members.

Users at Netflix are the administrators who are in charge of changes that happens in the application, the new releases and checking that everything is fine. Other ways include if the subscriber experiences issues with billing, streaming quality or account matters they have to contact the customer support. A visitor comes to the Netflix web site or to the Netflix application on a mobile device and scrolls down the list of offers. They can watch some trailers, read descriptions as well as sort films according to the type, for example, action, comedy or documentary and others. However, they will have to subscribe to any stream they desire, if they are to watch any stream at all.

The visitor subscribes by entering the first name and email, then an individual password, and lastly selecting one of the subscription plans: Basic, Standard or Premium. While using the Internet or social networks, or after registering to access higher level content after providing credit card or PayPal details, the person becomes a subscriber.

They were a subscriber and now they have the ability to watch anything that Netflix has to offer. They can search for the available shows or movies and mark them in the liste, for viewing them and can also watch the available programs on the mentioned devices such as smartphones, tablet, smart-tv, and computers etc. It uses a

recommendation engine where Netflix proposed materials the subscriber seems to be interested in because of what he has been viewing recently.

Clients have the rights to modify, the type of subscription plan they are using, alter payment methodology or revoke the service provided at any time. That includes the opportunity to create several profiles within one account to provide individual recommendations and watched TV shows for different users– family members.

Users at Netflix are the administrators who are in charge of changes that happens in the application, the new releases and checking that everything is fine. Other ways include if the subscriber experiences issues with billing, streaming quality or account matters they have to contact the customer support.

Key Steps:

1. Visitor Browses Content – A visitor explores the Netflix catalog but cannot watch full content until they subscribe.
2. Account Creation – The visitor registers for an account, selects a subscription plan, and provides payment information.
3. Subscriber Access – The new subscriber can now stream content, create watchlists, and receive personalized recommendations.
4. Account Management – The subscriber can change plans, update payment info, and access multiple profiles.
5. Platform Management – Netflix administrators update content and handle customer issues.

Actor	Brief Description
Subscriber	A user who subscribes to Netflix, browsing and watching content on the platform.
Admin	An employee who manages content uploads, user queries, and overall platform operations.
Content creator	An individual or organization that produces content such as movies or series for Netflix.
Content manager	A Netflix employee responsible for organizing, acquiring, and managing Netflix's streaming content.
Customer support	An employee who assists users with account-related issues, billing, or technical difficulties.
Payment sysytem	The system that processes payments and handles subscription renewals or changes.
Recommendation system	,A system that uses algorithms to suggest content to subscribers based on their viewing habits.
Advertiser	Businesses or entities that use Netflix to place advertisements (in regions with ad-supported plans).

Done by

Denil Tony

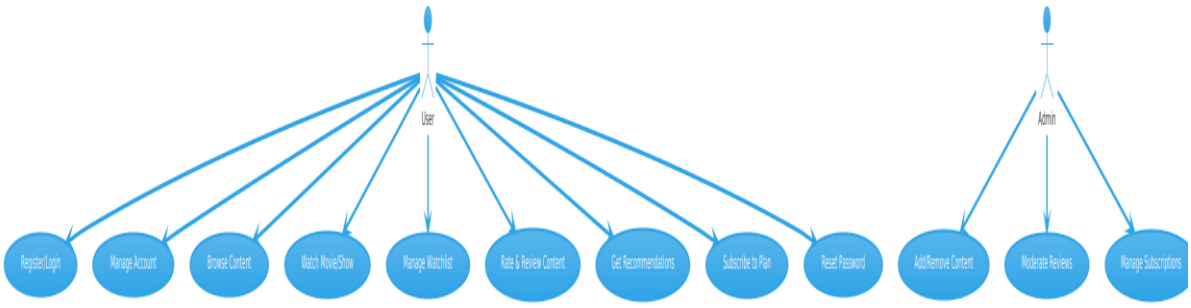
UML DIAGRAMS

Submitted by DENIL TONY

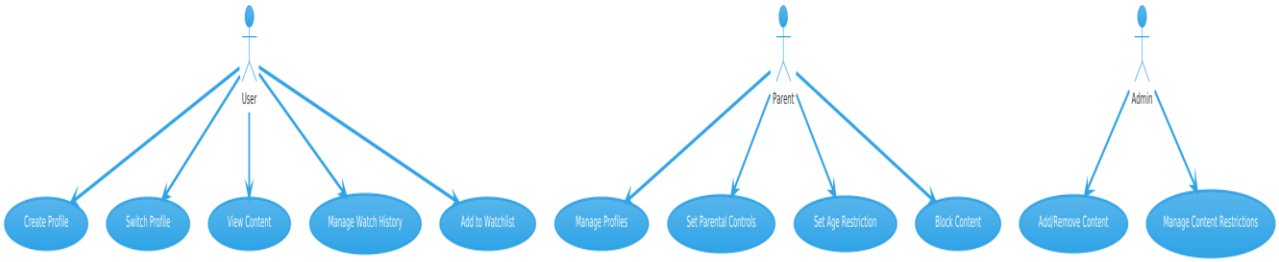
Use case diagram for Netflix

UML use case describes the functional requirements of a system and illustrates how the stakeholders utilise the system to perform certain tasks. A use case describes a specific behaviour of the system or each of the use cases refers to a function (i.e. ‘Log In,’ ‘Make Payment’) and it is associated with actors who may be the user or other systems that interact or trigger the function. These institutions are like include and extend model of defining responsibilities, differentiate between include and extend that the former suggest the same type of functionality and the latter suggests optional factors. From the Use case diagram again we can see the concept of what the system should deliver and acts as a framework to follow during design, implementation as well as testing.

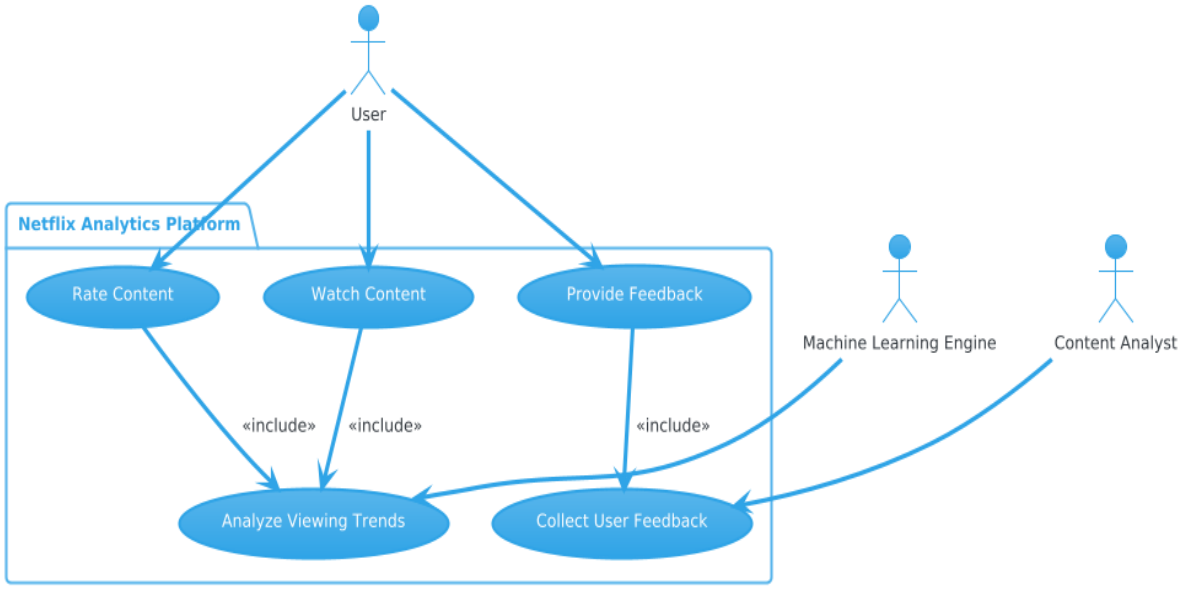
(Netflix use diagram)



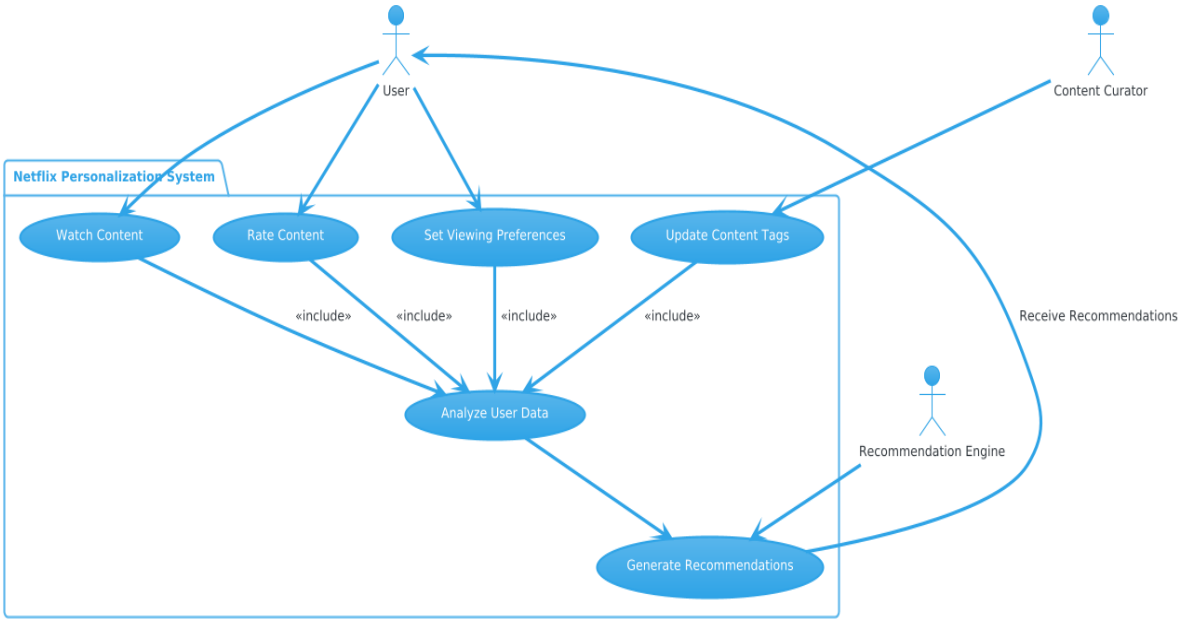
(user profile and parental controls)



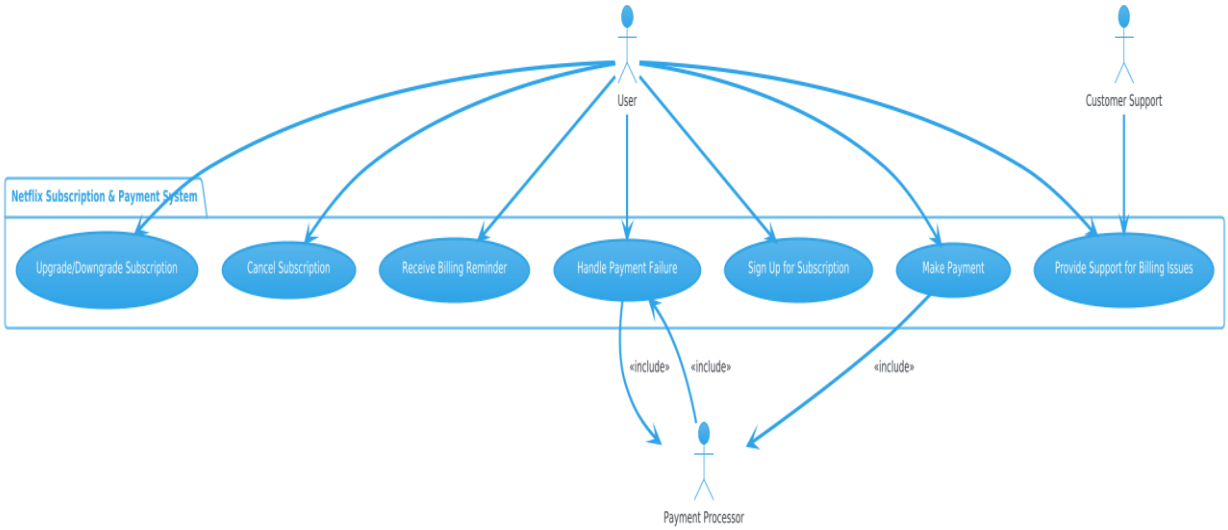
(content analytics and feedback management)



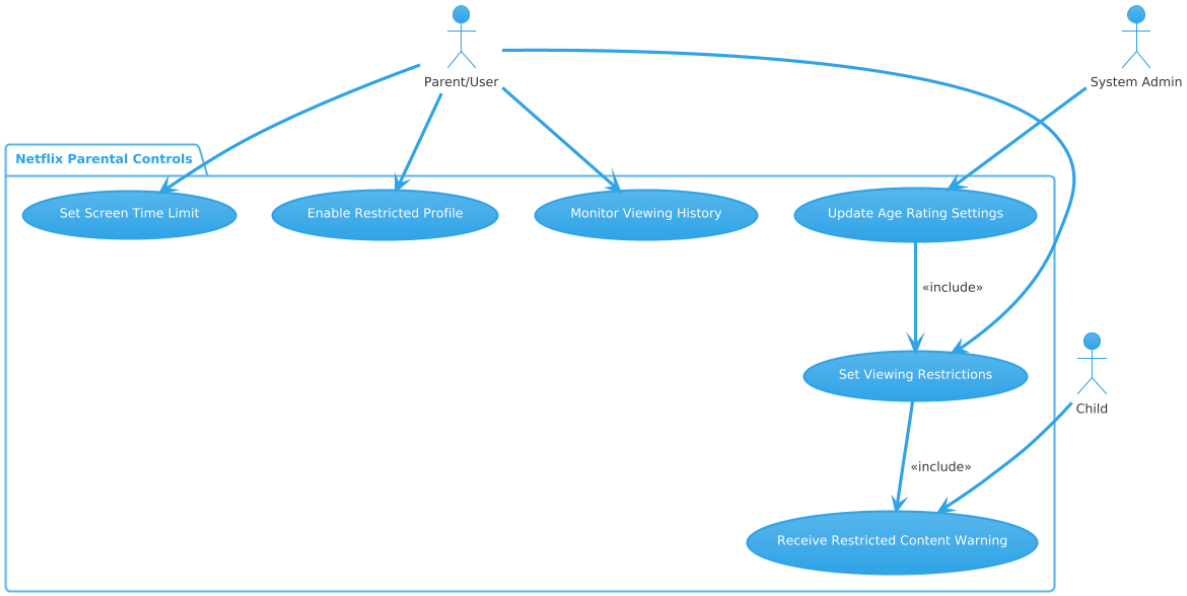
(Recommendation and personalisation system)



(Subscription and payments)



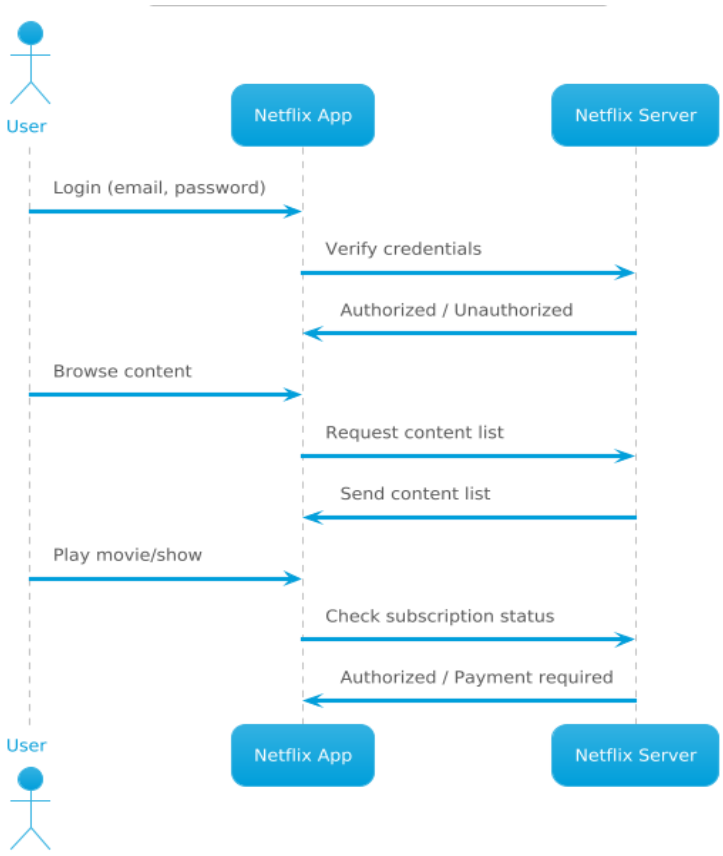
(controls and viewing restriction)



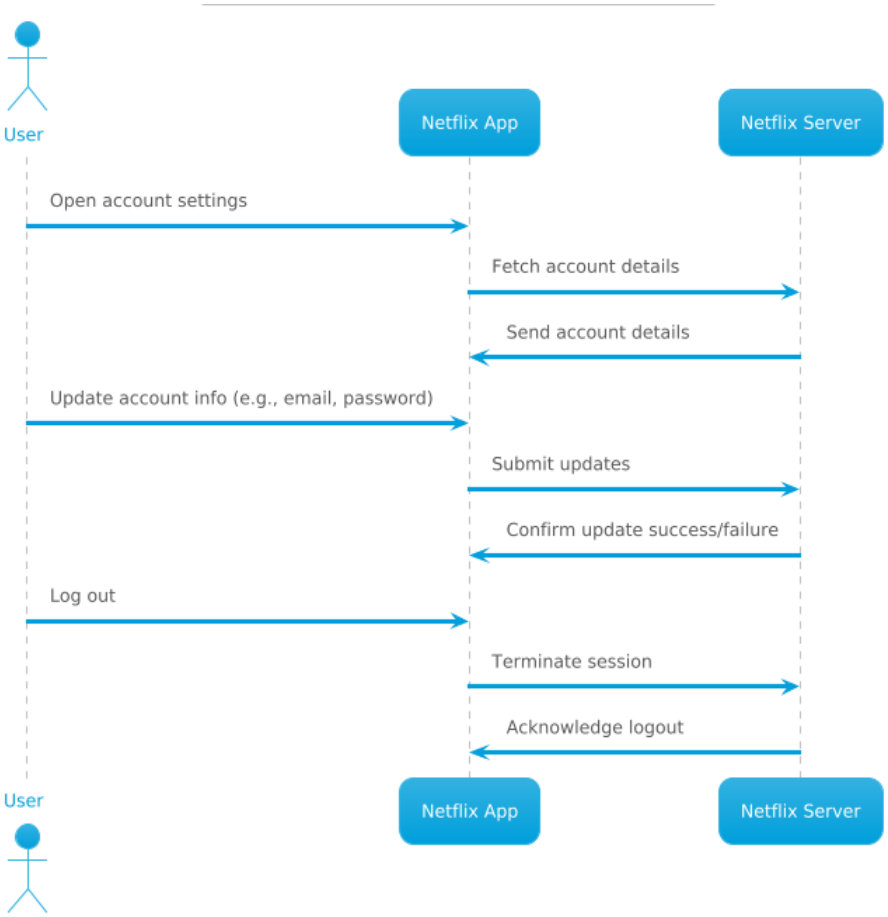
Sequence diagrams of Netflix

Sequence Diagrams are used in descriptions of patterns of communication over the time between objects or components of the system. These diagrams are most useful in knowing more about the behaviours of the system depending on user inputs as well as coordination of various components in executing a particular function.

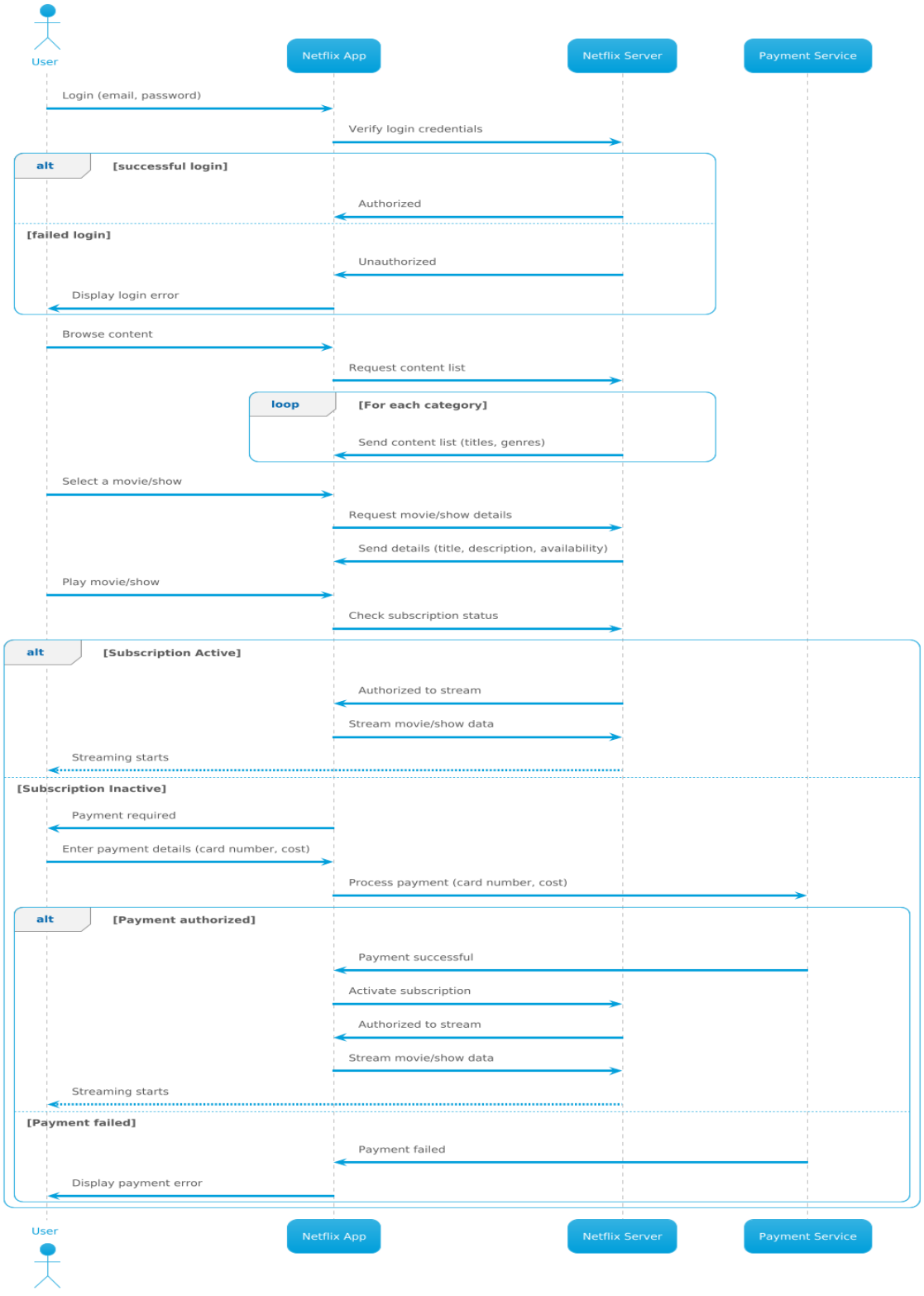
(Netflix basic diagram)



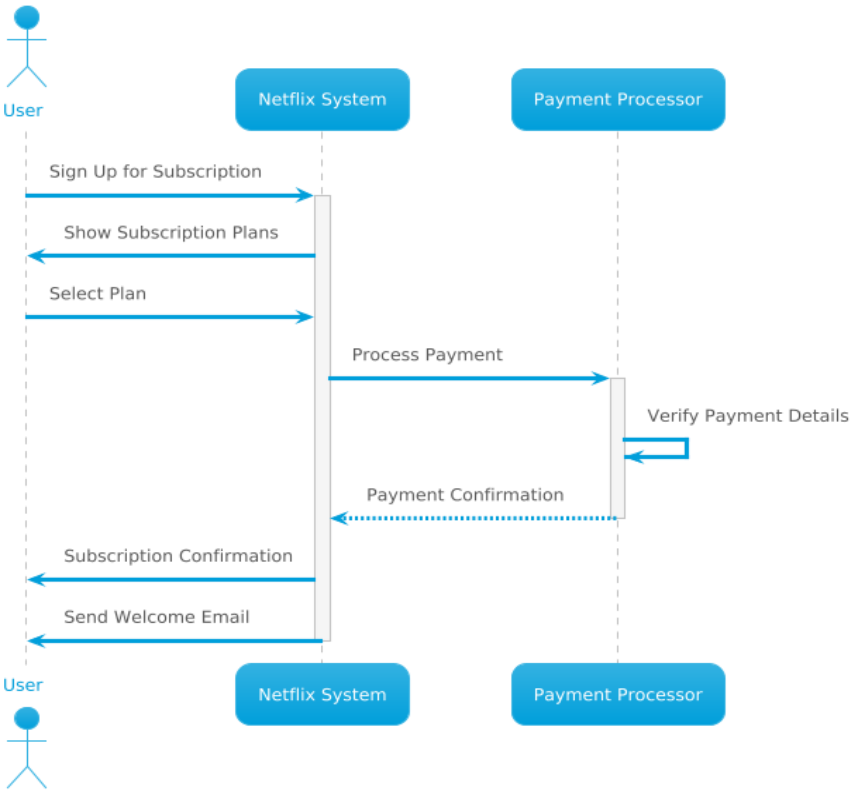
(Netflix account management)



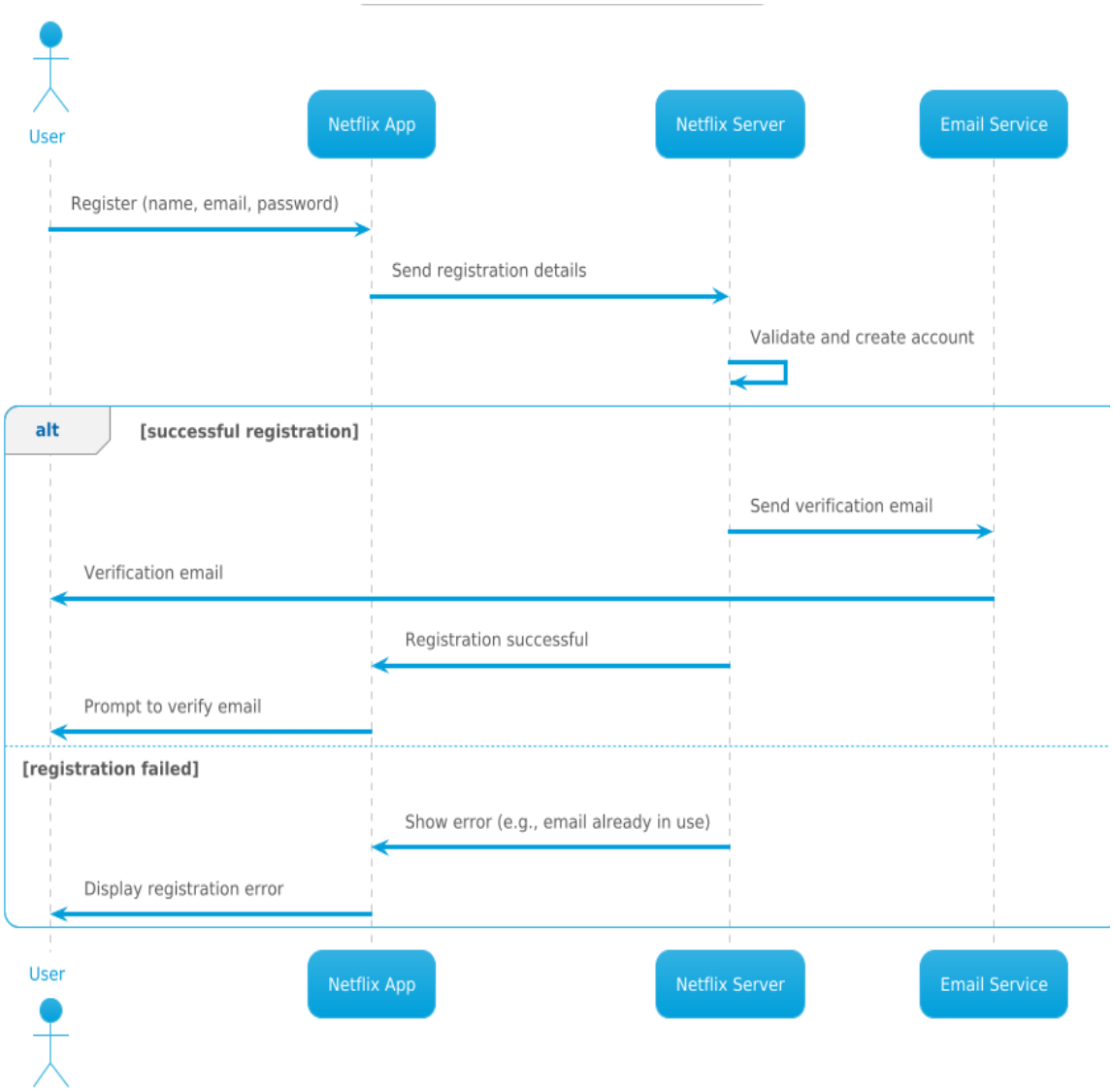
(Netflix streaming service)



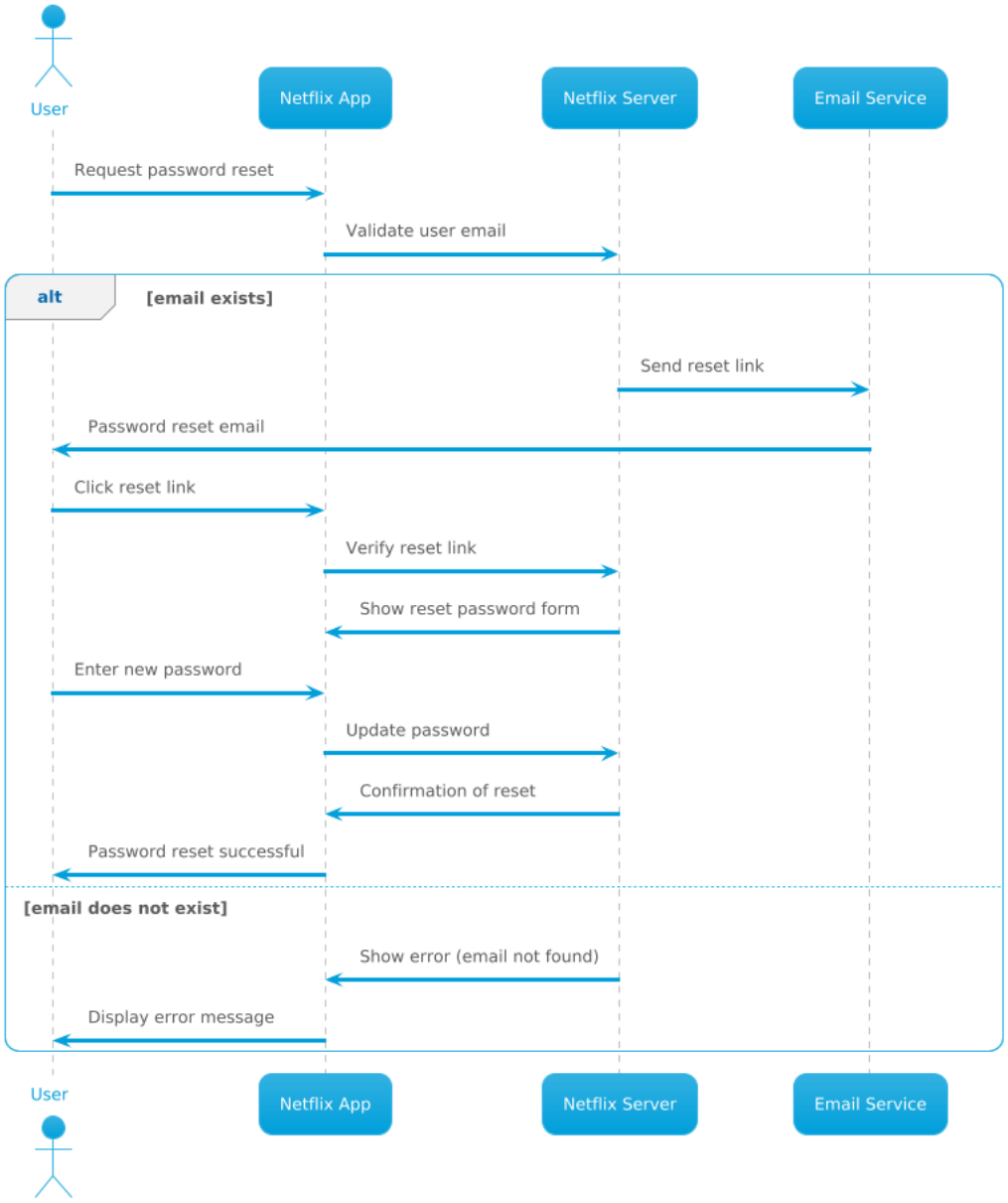
(Payment and subscription)


User
Netflix System
Payment Processor
Sign Up for Subscription
Show Subscription Plans
Select Plan
Process Payment
Verify Payment Details
Payment Confirmation
Subscription Confirmation
Send Welcome Email
User
Netflix System
Payment Processor

(Netflix user registration)



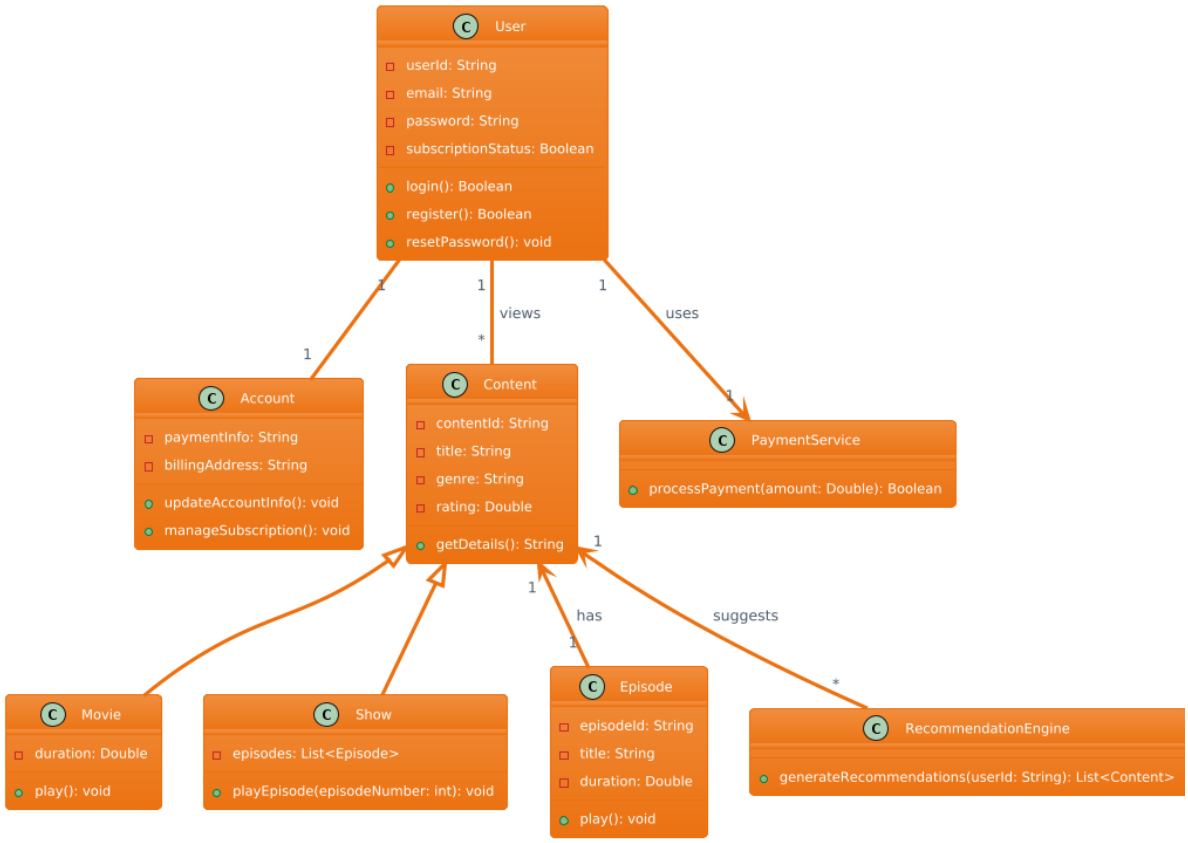
(Netflix password reset)



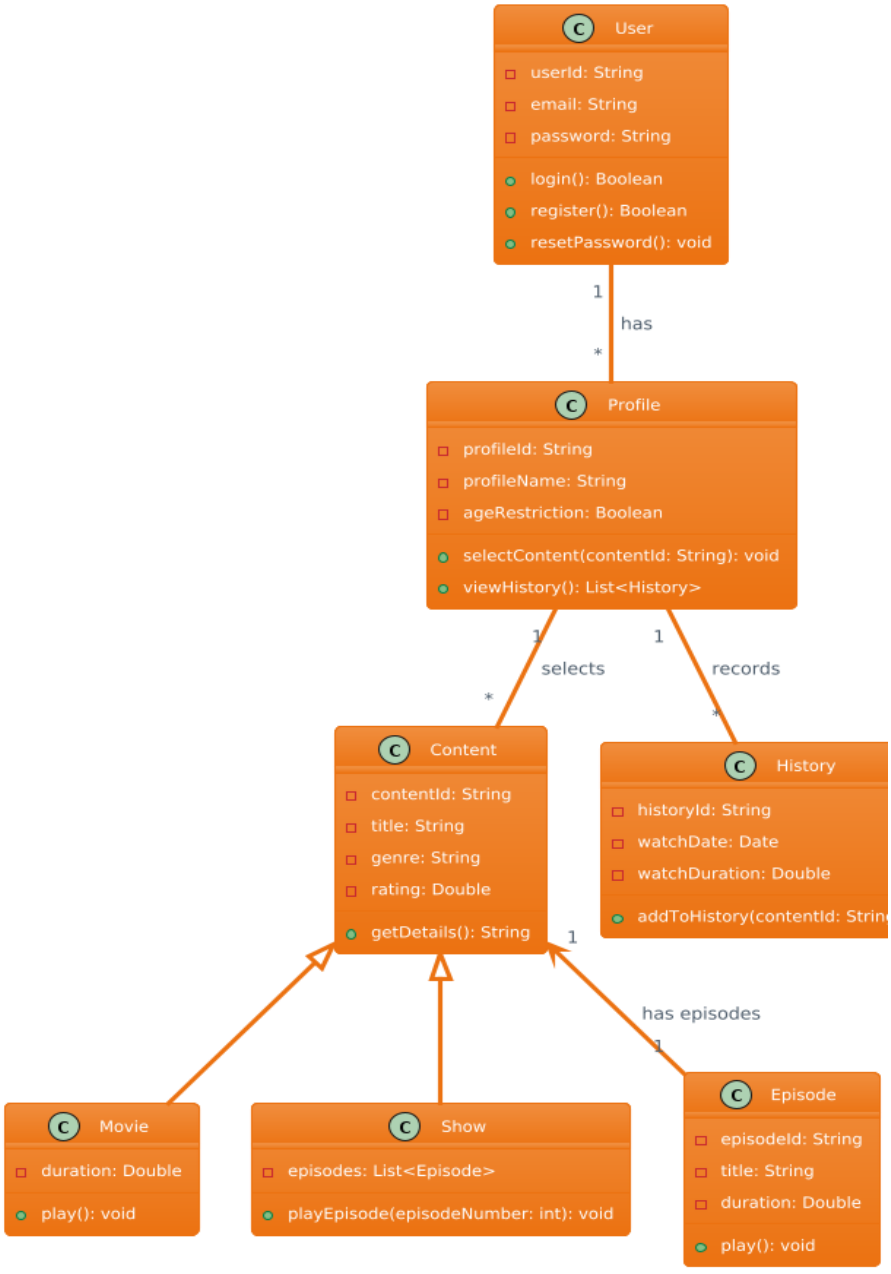
Class diagram of Netflix

A class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

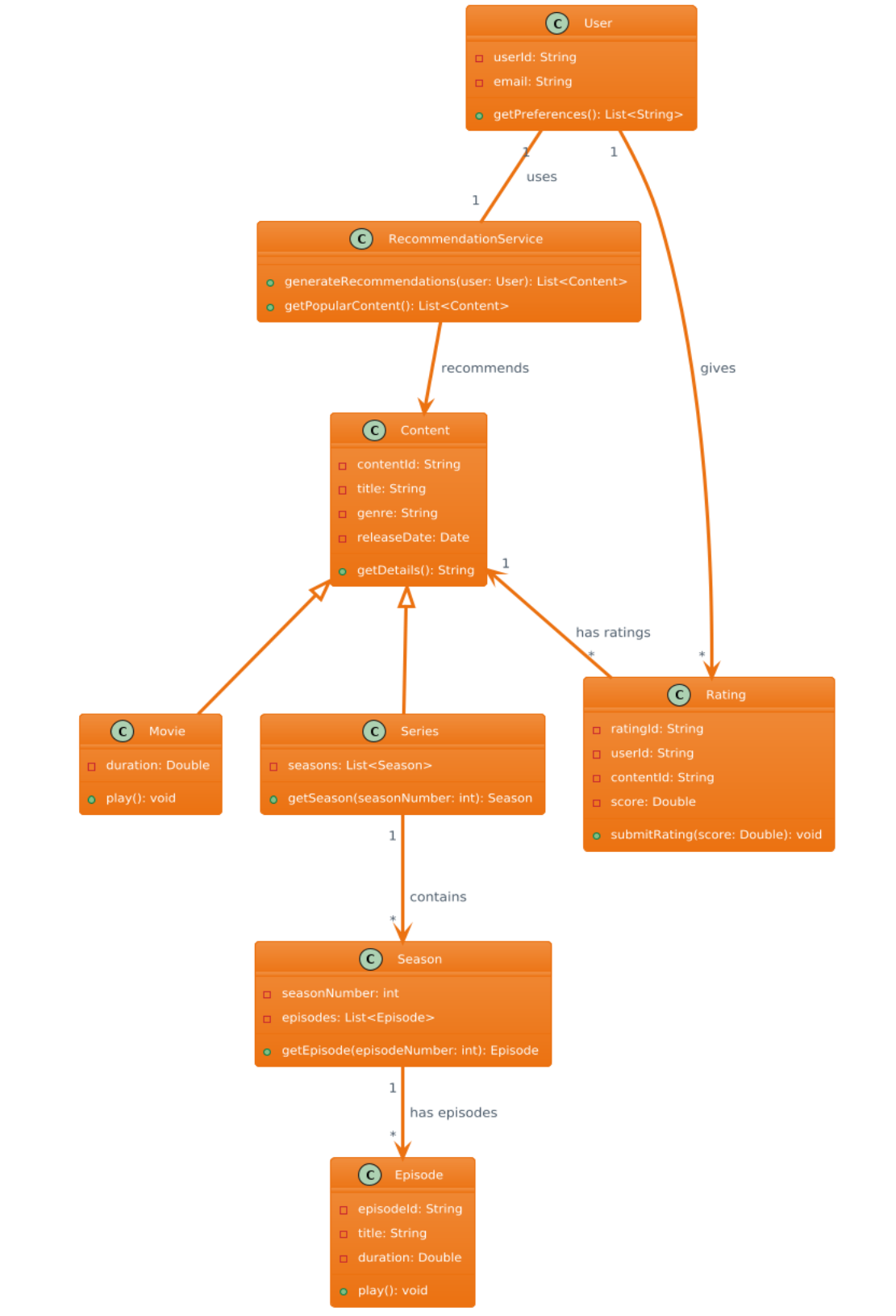
(Netflix class diagram)



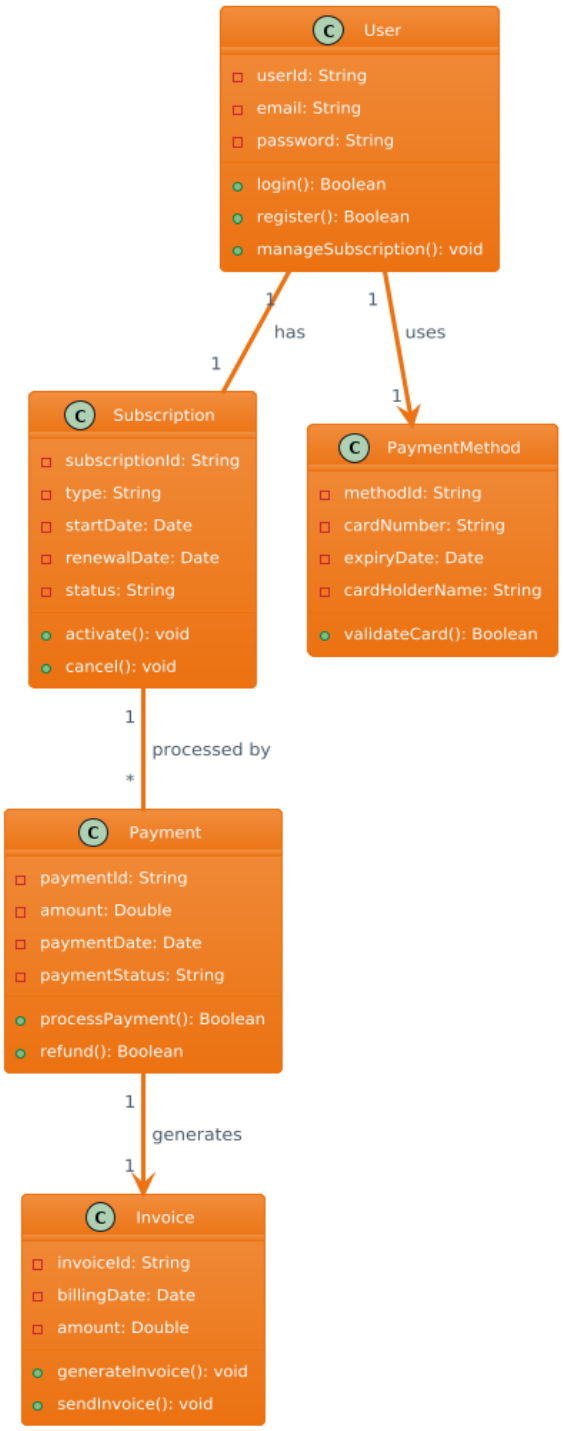
(Profiles and viewing history)



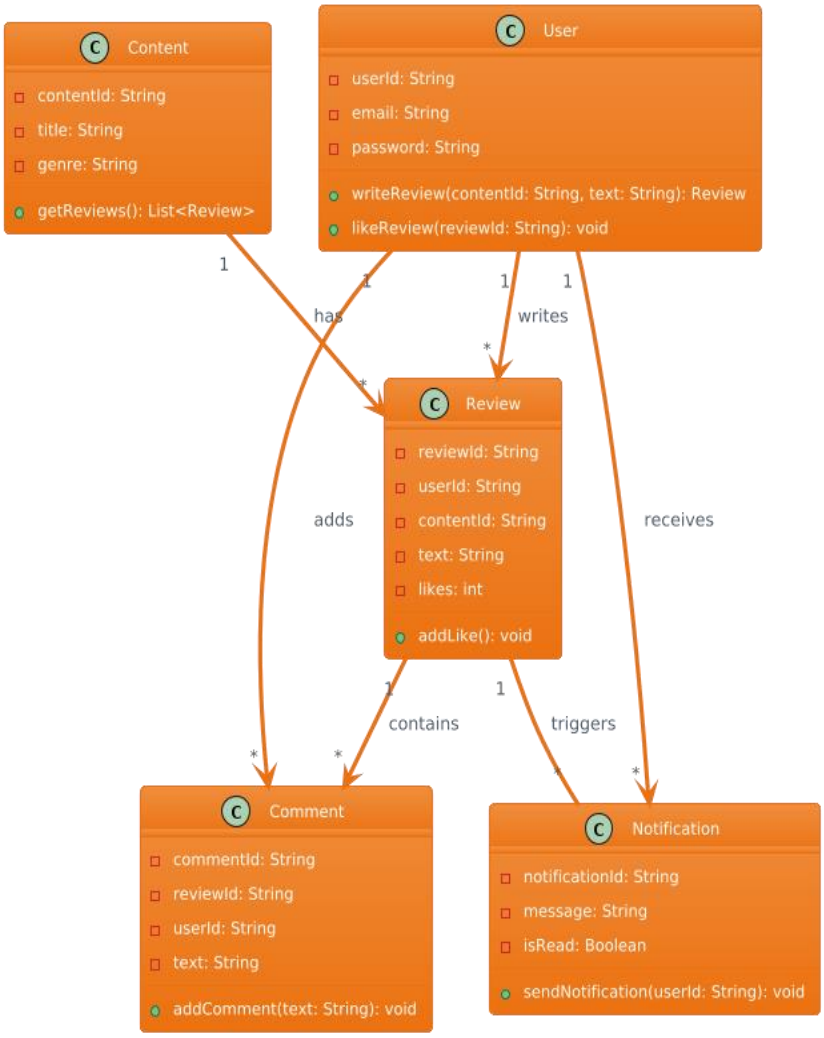
(Content and recommendations)



(Subscription and payments)



(Content review and user interaction)



(Watchlist and content categories)