

Product Functions

The VRS allows customers to search the video inventory provided by this video store. To rent videos through the VRS, one must register as a member using the VRS. Upon becoming a member and logging into the VRS, the VRS provides the functionality for renting videos, modifying membership information, and paying overdue fines.

The clerks of the video store use VRS to process the return of rented videos. The owner of the video store uses VRS to add new videos into the system, remove videos from the system, and modify video information.

User Characteristics

The three main groups of VRS users are customers, members, and store personnel. A customer is anyone who is not a member. The customer can only search through the video inventory. The amount of product training needed for a customer is none since the level of technical expertise and educational background is unknown. The only skill needed by a customer is the ability to browse a website.

A member is someone who has registered with VRS. A member can rent videos and pay fees online. As with a customer, these activities require no product training since the level of technical expertise and educational background of a member is unknown. The only skill needed by a member is the ability to browse a website.

The store personnel are divided into two groups: the clerk-level personnel and owner-level personnel. Their educational level is unknown and both group needs little to no training.

Functional Requirements

Customer subscribes for membership by providing personal information and credit/debit card information

The system shall provide registration for customers to become members.

The system shall assign a member number and allows the member to select a password and record membership information for future use.

The system shall allow the customer or member to search video information to help make good decision. Customer can browse the product and can add a movie to be rented into a wish list

The system shall provide login security for renting videos and payments.

The system shall allow members to rent videos.

After watching, the customer can send it back.

After receiving it the system will send another movie form the wish list.

The system shall allow a clerk to process returned videos.

The system shall email due notices to members. This will be done one day before due date.

The system shall send overdue notices every third day after the due date.

The system shall email a 60-day notice of charges for non-returned videos.

The system shall interface with the owner's email software.

The system shall allow the owner to add new videos to the video inventory.

The system shall retrieve video information from the video distributor.

The system shall interface with the video distributors' websites.

☐ The system shall allow the owner to remove videos from the video inventory.

☐ The system shall allow the owner to modify the video type.

Non-Functional Requirements

☐ The system shall provide web-access for all customer and membership functions.

☐ The system shall provide persistent storage for membership, rental, and video inventory information.

☐ The system shall provide an intuitive user interface that requires no training.

☐ The system shall provide a secure environment for financial transactions and for the storage of confidential member information.

. Wrong system requirements. System requirements are the premise on which systems are built. Just like the premise of every building is its foundation and the very strong buildings have very good foundations, so does it relate to systems development to. Most failed systems have come about because the organizations themselves do not know what they want.

Wrong system requirements.

System requirements are the premise on which systems are built. Just like the premise of every building is its foundation and the very strong buildings have very good foundations, so does it relate to systems development to. Most failed systems have come about because the organizations themselves do not know what they want.

As the initiating party, an organization must specify to developers the requirements to which the system must be built, taking into consideration their entire business process and the portions if not the entire process that the system is needed to augment, and communicate this need to developers in clear, precise and concise language devoid of ambiguity for the project to fit purpose. The reality though, is that, organizations are not being diligent enough to make a full and correct assessment of their processes and adequately determine their requirement and end up communicating half-truth requirement to developers. The system ends up solving only a fraction of what is expected of it and this complicates the processes ahead.

Many organizations also outsource the duty of requirement determination to the system developer and the outcome is often disastrous. This has also contributed to the determination of wrong requirements leading to the development of failed or challenged systems because, the interpretation the developer might have of your processes could be completely different from the organization's and if not resolved before development, the system would have been dead even before development.

2. Wrong scoping. Just like the old teaching that don't ever lie because, you will tell another lie to cover the first lie is exactly the scenario here. The scope of the system is determined on the requirements provided and if the requirement is faulty, the scope can never be faultless. This translates to the revision of scope which mostly results in cost sky-rocketing while leaving more and more demands unmet.

Scoping of project means assessing the requirements and determining the size of work to be done to fulfill consumer requirements. Scoping entails information gathering, consultation, discussion with stakeholders and monitoring of existing events. All of this is done to determine the boundary of work – what must be achieved. If all of the preceding functions are premised on an inaccurate requirement, the system thereof will not be fit for purpose. This is one of the key reasons why IT systems are challenge or fail.

3. Wrong feasibility study. After scoping to determine the scale of work to be done, developers together with the consumer need to make a feasibility analysis to determine whether the system can be delivered, putting in mind economic, political and technological and all relevant factors internally and externally, which can change anytime and have a bearing on the system development. The feasibility study answers the question of whether the project is attainable or not and how the project scope can be implemented to fit purpose for consumer satisfaction. This critical decision is premised on the scope done and scope is premised on the requirement given.

Software failures in the world

Poor communication between developers and customers. To be able to develop a system that fits purpose, there must be enough or even more than enough communication between the developers and customers. There must be a project team which comprise of key professionals from both ends who need to work hand in hand at every stage to birth a successful system. This does not always happen. After initiating the project, most customers wash their hands of it and leave it to the developers alone to figure out. No project team is setup to liaise with developers to guide and achieve milestones. This leaves developers to produce systems as they understand it which might not be a true reflection of what the customer wants and will result in a failed or challenged system which will either be redundant or partially fit for purpose.

3. Inadequate financing. Financing is an issue in every circumstance of life and business, given the limited availability of financial resource. This is what has prompted the desire for efficiency in business where managers want to do more and do it well with very little resource but this does not always work. Sometimes, organizations need to go all out with their expenses to achieve the best they want. Developing a system costs a fortune. Because, programmers are going to sacrifice their all to program the system and the revenue the system will aid the consumer to rake in in the long run will be in no comparison to the payment made for its purchase.

Also, other production factors like the size of the organization etc. drive up cost. Organizations normally want the best of systems to augment their processes but are not willing to spend good funds on them. Developers end up providing what they are paid for because if at the end of the day certain cost elements which will realize a successful system is not provided, they are constrained to make do with what they have and this results in a failed or challenged system most often

Myth #1: Most consumers want to have relationships with your brand.

Actually, they don't. Only 23% of the consumers in our study said they have a relationship with a brand. In the typical consumer's view of the world, relationships are reserved for friends, family and colleagues. That's why, when you ask the 77% of consumers who don't have relationships with brands to explain why, you get comments like "It's just a brand, not a member of my family." (What consumers really want when they interact with brands online is to get discounts).

How should you market differently?

First, understand which of your consumers are in the 23% and which are in the 77%. Who wants a relationship and who doesn't? Then, apply different expectations to those two groups and market differently to them. Stop bombarding consumers who don't want a relationship with your attempts to build one through endless emails or complex loyalty programs. Those efforts will be low ROI. Chances are there are higher returns to be had elsewhere in your marketing mix.

Myth #2: Interactions build relationships.

No, they don't. Shared values build relationships. A shared value is a belief that both the brand and consumer have about a brand's higher purpose or broad philosophy. For example, Pedigree Dog Food's shared value is a belief that every dog deserves a loving home. Southwest Airlines' shared value revolves around the democratization of air travel.

Of the consumers in our study who said they have a brand relationship, 64% cited shared values as the primary reason. That's far and away the largest driver. Meanwhile, only 13% cited frequent interactions with the brand as a reason for having a relationship.

