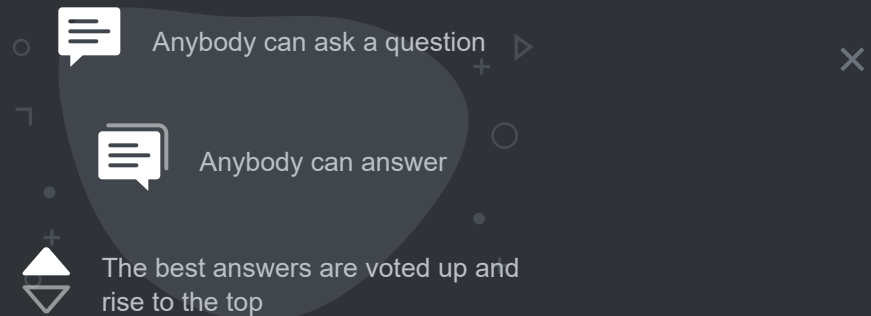


Software Engineering Stack Exchange is a question and answer site for professionals, academics, and students working within the systems development life cycle. It only takes a minute to sign up.

Join this community



## What is the difference between business and application logic? [closed]

Asked 4 years, 7 months ago   Active 4 years, 7 months ago   Viewed 12k times



11



3



**Closed.** This question needs [details or clarity](#). It is not currently accepting answers.

**Want to improve this question?** Add details and clarify the problem by [editing this post](#).

Closed 4 years ago.

Please note that I asked the same question on [stackoverflow](#) but they directed me to ask here.

While I am trying to discern the difference between the application logic and business logic I have found set of articles but unfortunately there is a contradiction between them.

[Here](#) they say that they are the same but the answer [here](#) is totally different.

For me I understand it in the following way:

If we look up for the definition of the `Logic` word in Google we will get

system or set of principles underlying the arrangements of elements in a computer or electronic device so as to perform a specified task.

So if the logic is `set of principles underlying the arrangements of elements` then the business logic should be `set of principles underlying the arrangements of the business rules`, in other words it means the rules the should be followed to get a system reflects your business needs.

And for me the application logic is `the principles that the application based on`, in other words, how to apply these rules to get a system reflects your business needs, for example should I use MVC or should not I use?, should I use SQL or MSSQL?.

So please could anybody help me to get rid of confusion about the difference between the application and the business logic.

business-logic

business-rules

edited May 23 '17 at 11:33



Community ♦

1

asked Aug 17 '15 at 11:43



user3260672

397 ● 2 ● 3 ● 8

- 3 The answers below the second SO link ([stackoverflow.com/questions/1456425/...](https://stackoverflow.com/questions/1456425/...)) you gave are correct and comprehensive. In short, they say "Business logic" is a subset of "Application logic". – Doc Brown Aug 17 '15 at 13:29 ✎
- 2 ... and the encyclopedia2 link you gave tells IMHO the same, so where is your problem? – Doc Brown Aug 17 '15 at 13:35
- 1 ... and since you decided not to add any clarification, I am voting to close as "unclear what you are asking": – Doc Brown Aug 17 '15 at 20:36

@DocBrown, I have edited the title of the question and the last paragraph in the question, but I think that it was clear what I wanted and I got the answer!! have you read the question??. – user3260672 Aug 19 '15 at 9:12

## 4 Answers

Active

Oldest

Votes

▲ I agree with SO's LoztInSpace that this is quite opinionated answer and that everyone can have slightly different definitions. Especially if historical influences are involved. This is how I would define the terms:

13



Business logic is logic, that is created with collaboration and agreement with business experts. If business expert says that "Customer cannot withdraw more money than he has in his account.", then this is a business rule. In ideal world, this logic would be in some kind of library or service, so it can be either reused across multiple applications or changed in all relevant applications at once.



Application logic is simply everything else. Example can be "clicking this button opens window to add new customer". It has nothing to do with business, but it is still logic that needs to be implemented. In ideal world, application logic will use library or service, that is implementing the business rules. Multiple application, each with different application logic, can reuse one business logic. Imagine web app, web service and mobile app all operating using one business logic, but each clearly need different application logics.

The reason why I think those two get mixed up, is that keeping them separate is extremely hard. Even if you do your most to keep them separate, use cases surface where you have to mix them up. If for example you have all your business logic in service, it keeps it separate. But having some business logic in local application that is using the service might increase responsiveness or user comfort, because the local application doesn't need to call service for every small change.

Another reason why they are mixed together is that for many non-technical people. UI is "the application", so anything reflected in the UI is important. In the ideal "business logic" case, there is no UI. There would probably be suite of automated tests to verify the logic, but nothing that can be shown to business people. So to business people, everything is same kind of "logic". IMO.

answered Aug 17 '15 at 13:00



Euphoric

31.6k ● 6 ● 61 ● 90

---

1 "Application logic is simply everything else" - to be nitty, I would say "it includes business logic and everything else" (but it probably depends on whom you talk to) – [Doc Brown](#) Aug 17 '15 at 13:31

---



3



As others have pointed out, these terms do not have one universally accepted meaning. I will describe the definitions I have encountered more often, i.e. in several projects with different companies.

The **business logic** defines a normalized, general-purpose model of the business domain for which an application is written, e.g.

- Classes like `Customer` , `Order` , `OrderLine` , and associations like `customer-order` , and so on.
- General-purpose operations such as `registerCustomer` , `cancelOrder`

Very often this class model is mapped to a database model and the mapping is implemented using ORM. The operations are normally performed each in their own transaction and provide the basic API for modifying the database, i.e. the persistent state of the application.

The **application logic** is a layer built on top of the business logic and serves to implement specific use cases. Application logic modules may use ad-hoc data representation, e.g. a `CustomerSummary` class without any association to `Order` if you want to list customers only. Such ad-hoc data representation must be mapped to the underlying normalized representation provided by the business model. For example, `CustomerSummary` can be defined as a view on top of `Customer` .

Note that the boundary between the two layers may not be so clearly-defined. E.g. after implementing several use cases one might notice similar data structures in the application logic and decide to unify (normalize) them and move them to the business logic.

answered Aug 17 '15 at 17:32



Giorgio

18.1k ● 14 ● 71 ● 129



Every system or application is going to have its own definitions of what is business logic and what is application logic. It will either be explicit or implicit.

2

In my experience data driven applications (e.g. DBs etc.) tend to have a more formal definition of what the business logic is.



The application logic tends to focus on getting information from point A to point B, the business logic centres around what the information is - and the language of the business logic is usually domain specific. Put another way, the application logic is focused on the question "how does it work?", the business logic on "what does it do?" - again, the distinction can be very fuzzy and is more often that not domain specific.

edited Aug 17 '15 at 17:42

answered Aug 17 '15 at 13:53



Niall

1,691 ● 1 ● 14 ● 19



1

Na, they're just different terms for the same thing - the "middle tier" of program code that does the things you want your program to perform. Like many things in software, there are no hard-and-fast terminology for pieces of a system, as there are no single formal definitions for building systems.



So sometimes people will call it business logic, others application logic, others will call it program logic, its all much of a muchness. Don't bother trying to define this so rigidly, nearly every system varies in how its built so be glad there's only this minor level of vagueness in terminology!

answered Aug 17 '15 at 13:19



gbjbaanb

46.7k ● 6 ● 94 ● 165