



Off-the-Shelf vs. Bespoke Software

Off-the-shelf software is software that is ready-to-use and is being marketed as a product by a software company, whereas bespoke software is software that is custom-built by a software development company.

Advantages of Off-the-Shelf Software

Price

Off-the-shelf software is generally cheaper than custom-built software as the development costs are spread amongst many customers.

Reliability

Off-the-shelf software has been tried and tested both by the seller and by existing customers.

Readily Available Support

There will generally be a wealth of support documentation available as there are lots of other users using the same software. You may also be able to find third-party training courses and discussion groups on the Internet.

Available Now

You do not need to spend the time specifying the software and then waiting for it to be developed. It will be ready for you to use straight away.

Disadvantages of Off-the-Shelf Software

Customisation May Not Be Possible

Off-the-shelf software is usually designed to be as flexible as possible so that it can suit the needs of the largest number of potential users. However, you may require the software to perform a task that is specific to your business which it is unable to do. It is unlikely that you will be able to get the publishers of the software to make any customisations for you.

Change Business Processes

If the software does not exactly fit your business processes you may be forced to adapt your business to fit the software.

Overly Complex

As the software is designed to accommodate a variety of users, it is likely that there will be large sections of the software that you are paying for but are not relevant to your organisation. You may also find that these sections get in the way when you are trying to use the rest of the system.

Little Competitive Advantage

As your competitors can buy and use the same software, you will gain little competitive advantage from off-the-shelf software.

Advantages of Bespoke Software

Exactly Meets Your Requirements

As the software has been custom-built for you it should exactly fit your requirements and should integrate well with your business processes. This is the main reason for organisations choosing bespoke software.

More Intuitive For Users

The users of the system will most likely already be used to the way your organisation works. Therefore if the software fits in with this methodology it will be easier for them to start using it.

Competitive Advantage

You may be able to develop software that performs a task that your competitors' IT systems do not, thus giving you a competitive advantage. Also, if you own the intellectual property for the bespoke software you have the option to sell the application to other businesses and recoup some/all of your investment.

Future-Proof

As your organisation evolves the software can be modified and extended to suit your needs.

No Licensing Costs

Generally, bespoke software will just involve one-off payments for the development of the system; therefore you do not need to pay ongoing licensing costs as you may have to do with off-the-shelf software.

Disadvantages of Bespoke Software

Cost

Bespoke software will generally be more expensive to start using as it will involve a large initial investment into its development. However, once paid for it, it may not cost you anything further to use it.

Difficulty Selecting a Good Developer

There are a lot of software development companies in the UK, some of which are great and some of which are very poor. [Selecting a good software developer](#) can be difficult, particularly if you have no experience in the software development process. If you choose a good software developer they will be able to produce a piece of software that will greatly enhance your business. However, if you choose a bad software developer it may cost you a lot of time and money to rectify the problems they have created.

Time Consuming

The development of bespoke software can take a long time to complete and may involve you allocating a lot of your time to the development process – e.g. writing the brief, meeting with the developers, testing the system, providing feedback.