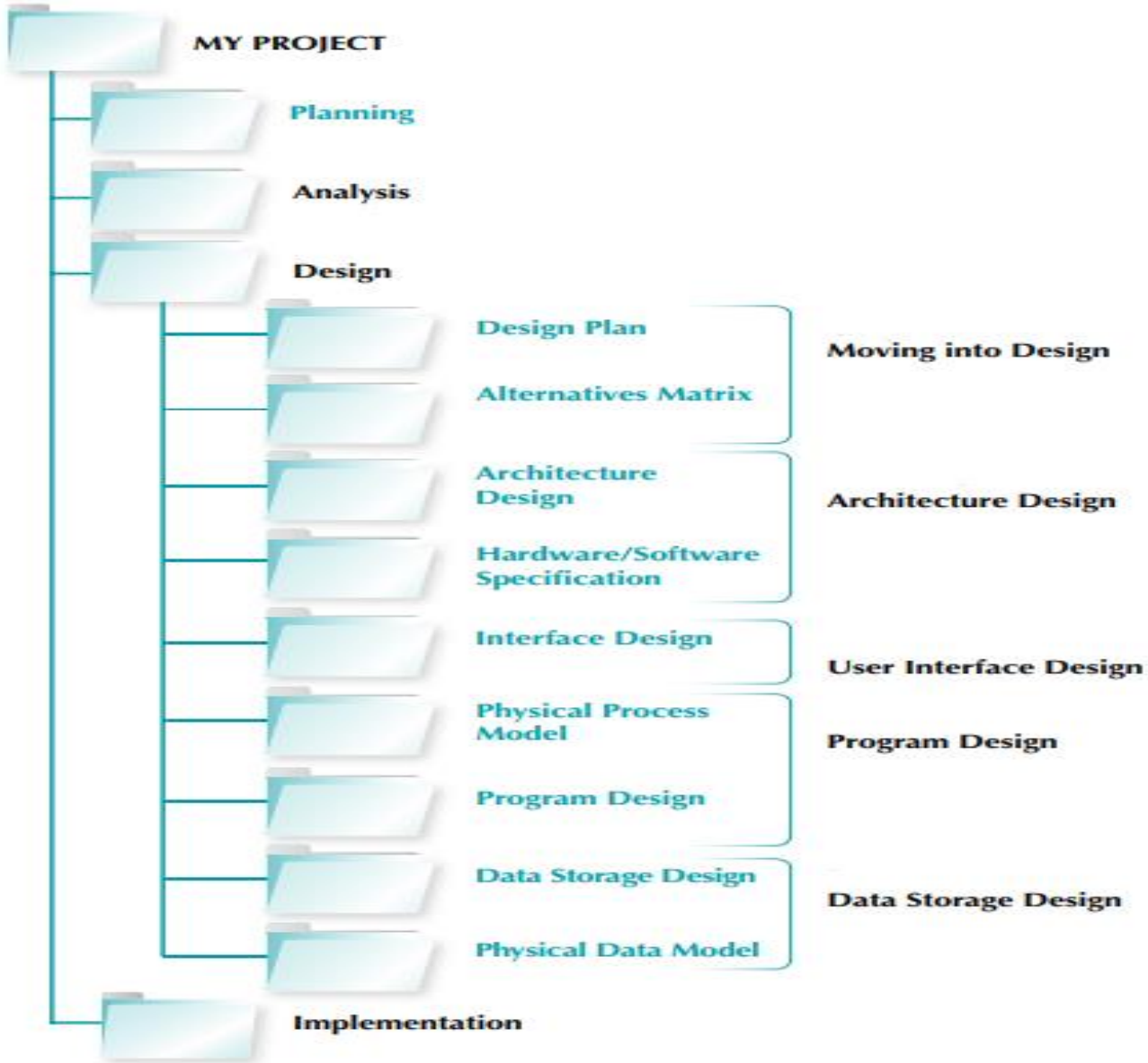
Transition from Requirements to Design


MY PROJECT
Planning
Analysis
Design
Design Plan
Alternatives Matrix
Moving into Design
Architecture Design
Hardware/Software Specification
Architecture Design
Interface Design
User Interface Design
Physical Process Model
Program Design
Program Design
Data Storage Design
Physical Data Model
Data Storage Design
Implementation

System design

The design phase decides how the new system will operate

The purpose of the design phase is to decide how to build it. System design is the determination of the overall system architecture—consisting of a set of physical processing components, hardware, software, people, and the communication among them—that will satisfy the system's essential requirement

During the initial part of design, the project team converts the business requirements for the system into **system requirements** that describe the technical details for building the system.

Unlike business requirements, which are listed in the requirements definition and communicated through use cases and logical process and data models, system requirements are communicated through a collection of design documents and physical process and data models. ***Together, the design documents and physical models make up the blueprint for the new system.***

Activities of the design phase.

Activities in the Design Phase	Deliverables
✓ Determine preferred system acquisition strategy (make, buy, or outsource).	– Alternative matrix
✓ Design the architecture for the system.	– Architecture design
✓ Make hardware and software selections.	– Hardware and software specification
✓ Design system navigation, inputs, and outputs.	– Interface design
✓ Convert logical process model to physical process model.	– Physical process model
✓ Update CASE repository with additional system details.	– Updated CASE repository
✓ Design the programs that will perform the system processes.	– Program design specifications
✓ Convert logical data model to physical data model.	– Physical data model
✓ Update CASE repository with additional system details.	– Updated CASE repository
✓ Revise CRUD matrix.	– CRUD matrix
✓ Design the way in which data will be stored.	– Data storage design
✓ Compile final system specification.	– System specification: all of the above deliverables combined and presented to approval committee

There are, however, actually three primary ways to approach the creation of a new system:

(1) develop a custom application in-house;
(2) buy a packaged system and (possibly) customize it;
(3) rely on an external vendor, developer, or service provider to build or provide the system.

Each of these choices has its strengths and weaknesses

Summary of software acquisition options pros and cons.

Pros	Cons
Custom Development	
Get **exactly** what we want New system built consistently with existing technology and standards Build and retain technical skills and functional knowledge in-house Allows team flexibility and creativity Unique solutions created for strategic advantage	Requires significant time and effort May add to existing backlogs May require skills we do not have Often costs more Often takes more calendar time Risk of project failure
Packages (purchased or obtained from ASP or SaaS)	
No need to "reinvent the wheel" for common business needs Tested, proven product Cost savings Time savings Utilize vendors' expertise Some customization may be possible	Rarely a perfect fit Organizational processes must adapt to software Reliance on vendor for maintenance and future enhancements Will not develop in-house functional and technical skills Unique needs may go unmet May require system integration
Outsourced Development	
Hire expertise we do not have May save time and money Lower risk	No opportunity to build in-house expertise Reliance on vendor Future options limited Security—potential loss of confidential info Performance based on contract terms

Software as a service (SaaS)

Custom Development

Many project teams assume that custom development, or building a new system from scratch, **is the best way to create a system**, because teams have complete control over the way the system looks and functions.

Packaged Software

Many business needs are not unique, and because it makes little sense to reinvent the wheel, many organizations buy packaged software that has already been written, rather than developing their own custom solution. In fact, there are thousands of commercially available software programs that have already been written to serve a multitude of purpose.

Outsourcing

The acquisition choice that requires the least in-house resources is outsourcing, which means hiring an external vendor, developer, or service provider to create or supply the system. Out- sourcing has become quite popular in recent years, with both US and non-US (offshore) service providers available.

The term **outsourcing** has come to include a variety of ways to obtain IT services and products. Outsourcing firms called **application service providers (ASPs)** supply software applications and/or software-related services over wide area networks or the Internet. In this approach to obtaining software, the ASP hosts and manages a software application, and owns, operates, and maintains the servers that run the application. The ASP also employs the people needed to maintain the application.

Software as a Service (SaaS) is a popular term that is essentially an extension of the ASP model. This term is commonly used to describe situations in which SaaS vendors develop and manage their own software rather than managing and hosting **a third-party independent soft- ware vendor's software** (the more traditional ASP model)

Outsourcing guidelines.

- Keep the lines of communication open between you and your outsourcer.
- Define and stabilize requirements before signing a contract.
- View the outsourcing relationship as a partnership.
- Select the vendor, developer, or service provider carefully.
- Assign a person to manage the relationship.
- Do not outsource what you do not understand.
- Emphasize flexible requirements, long-term relationships, and short-term contracts.

Influences on the Acquisition Strategy

There are valid reasons for choosing any of the acquisition strategies just discussed

	When to Use Custom Development	When to Use a Packaged System	When to Use Outsourcing
Business need	The business need is unique.	The business need is common.	The business need is not core to the business.
In-house experience	In-house functional and technical experience exists.	In-house functional experience exists.	In-house functional or technical experience does not exist.
Project skills	There is a desire to build in-house skills.	The skills are not strategic.	The decision to outsource is a strategic decision.
Project management	The project has a highly skilled project manager and a proven methodology.	The project has a project manager who can coordinate vendor's efforts.	The project has a highly skilled project manager at the level of the organization that matches the scope of the outsourcing deal.
Time frame	The time frame is flexible.	The time frame is short.	The time frame is short or flexible.

Business Need

If the business need for the system is common and technical solutions already exist in the marketplace that can fulfill the system requirements, it is usually appropriate to select a packaged software solution

Outsourcing can be used to assist a company with custom development projects and to acquire software packages.

Many organizations have experimented with using offshore outsourcing as a way of “exporting” IT-related work to countries that have lower labor costs. Good quality IT skills are available in a number of countries, but companies considering this option in order to save money need to carefully manage the risks of this way of obtaining IT services.

In-House Experience

If in-house experience exists for all the functional and technical needs of the system, it will be easier to build a custom application than if these skills do not exist.

For example, a project team that does not have Web commerce technology skills may want to acquire a Web commerce package that can be installed without many changes. Outsourcing is a good way to bring in outside experience that is missing in-house so that skilled people are in charge of building the system.

Project Skills

The skills that are applied during projects are either technical (e.g., Java, Structured Query Language [SQL]) or functional (e.g., electronic commerce), and different design alternatives are more viable, depending on how important the skills are to the company's strategy

Project Management

Custom applications require excellent project management and a proven methodology. There are so many things that can push a project off track, such as funding obstacles, staffing holdups, and overly demanding business users.

Time Frame

When time is a factor, the project team should probably start looking for a system that is already built and tested. In this way, the company will have a good idea of how long the package will take to put in place and what the final result will contain.

Selecting an Acquisition Strategy

Once the project team has a good understanding of how well each acquisition strategy fits with the project's needs, it must begin to understand exactly how to implement these strategies

For example, what tools and technology would be used if a custom alternative were selected? What vendors make packaged systems that address the project needs? What service providers would be able to build this system if the application were outsourced?

Project teams employ several approaches to gather additional information that is needed. One helpful tool is **the request for proposal (RFP),** a document that solicits a formal proposal from a potential vendor, developer, or service provider. RFPs describe in detail the system or service that is needed, and vendors respond by describing in detail how they could supply those needs.

For smaller projects with smaller budgets, **the request for information (RFI)** may be sufficient. An RFI is a shorter, less detailed request that is sent to potential vendors to obtain general information about their products and services.

When a list of equipment is so complete that the vendor need only provide a price, without any analysis or description of what is needed, **the request for quote (RFQ)** may be used.

Alternative Matrix

An **alternative matrix** can be used to organize the pros and cons of the design alternatives so that the best solution will be chosen in the end

Evaluation Criteria	Relative Importance (Weight)	Alternative 1: Custom Application Using VB.NET	Score (1–5)*	Weighted Score	Alternative 2: Custom Application Using Java	Score (1–5)*	Weighted Score	Alternative 3: Packaged Software Product ABC	Score (1–5)*	Weighted Score
Technical Issues:										
Criterion 1	20		5	100		3	60		3	60
Criterion 2	10		3	30		3	30		5	50
Criterion 3	10		2	20		1	10		3	30
Economic Issues:										
Criterion 4	25	Supporting	3	75	Supporting	3	75	Supporting	5	125
Criterion 5	10	Information	3	30	Information	1	10	Information	5	50
Organizational Issues										
Criterion 6	10		5	50		5	50		3	30
Criterion 7	10		3	30		3	30		1	10
Criterion 8	5		3	15		1	5		1	5
TOTAL	100			350			270			360

* This denotes how well the alternative meets the criteria. 1 = poor fit; 5 = perfect fit.

This matrix is created by the same steps as the feasibility analysis

Alternative matrix for website builder program

Evaluation Criteria	Relative Importance (Weight)	Alt 1: WB-1	Score (1-5)*	Wtd Score	Alt 2: WB-2	Score (1-5)*	Wtd Score	Alt 3: WB-3	Score (1-5)*	Wtd Score
Technical Issues:										
Integration with existing infrastructure	15	Very little capability	2	30	Provided, but appears awkward	3	45	Strong, appears seamless	5	75
Database capabilities	15	None	1	15	Limited	2	30	Excellent; compatible with company standards	5	75
Access to underlying code	10	Not possible	1	10	Limited	3	30	Easy	5	50
Video support	15	Yes; adequate	3	45	Yes; adequate	3	45	Yes; excellent	5	75
Economic Issues:										
Cost	20	$15/month	5	100	$25/month	4	80	$90/month	1	20
Organizational Issues:										
Market adoption	5	Strong - widely used	4	20	Moderate - newer product	3	15	Strong - market leader	5	25
Ease of learning	10	High	5	50	Somewhat complex	3	30	High learning curve	1	10
Ease of use	10	Inflexible	2	20	Somewhat flexible	4	40	Very flexible; easy to modify	5	50
TOTAL	**100**			**290**			**315**			**380**

* The score denotes how well the alternative meets the criteria; 1 = poor fit; 5 = perfect fit.