

UNIT 1

INTRODUCTION

1.1 Introduction to Software Development:

Software development is a complicated process. It requires careful planning and execution to Meet the goals. Sometimes a developer must react quickly and aggressively to meet ever-changing market demands. Maintaining software quality hinders fast-paced software development,as many testing cycles are necessary to ensure quality products.

Software development is a complicated process comprising many stages. Each stage requires a lot of paperwork and documentation in addition to the development and planning process. This is in contrast to the common thinking of newcomers to the software industry who believe that software development is just “writing code.” Each software development project has to go through at least the following stages:

- Requirement gathering
- Writing functional specifications
- Creating architecture and design documents
- Implementation and coding
- Testing and quality assurance
- Software release
- Documentation
- Support and new features

1.1.1 Requirement Gathering

Requirement gathering is usually the first part of any software product. This stage starts when you are thinking about developing software. In this phase, you meet customers or prospective customers, analyzing market requirements and features that are in demand. You also find out if there is a real need in the market for the software product you are trying to develop.

In this stage, marketing and sales people or people who have direct contact with the customers do most of the work. These people talk to these customers and try to understand what they need. A comprehensive understanding of the customers’ needs and writing down features of the proposed software product are the keys to success in this phase. This phase is actually a base for the whole development effort. If the base is not laid correctly, the product will not find a place in the market. If you develop a very good software product which is not required in the market, it does not matter how well you build it.

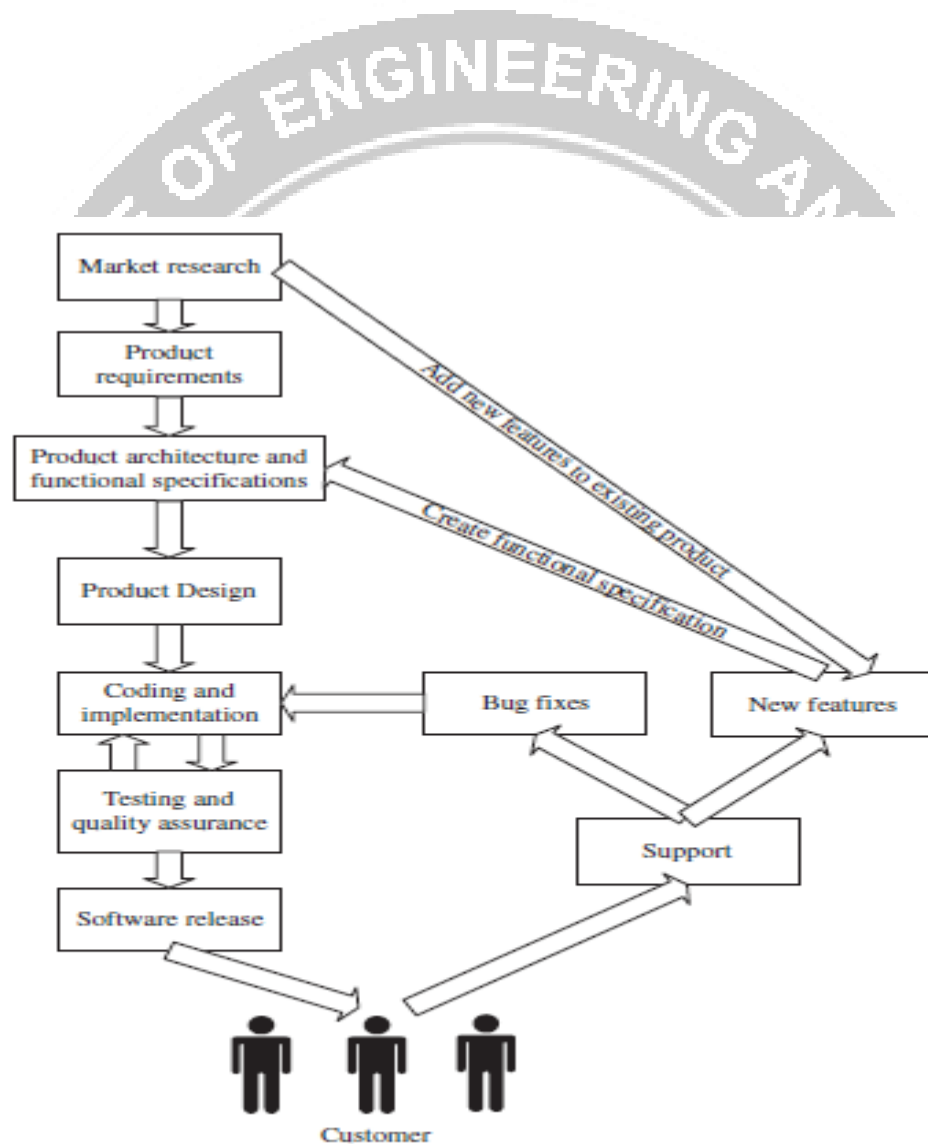


Figure 1-1 Typical processes for software development projects.

1.1.2 Writing Functional Specifications

Functional specifications may consist of one or more documents. Functional specification documents show

the behavior or functionality of a software product on an abstract level. Assuming the product is a black box, the functional specifications define its input/output behavior. Functional specifications are based upon the product requirements documentation put forward by people who have contact with the enduser of the product or the customers. In larger products, functional specifications may consist of separate documents for each feature of the product. For example, in a

router product, you may

have a functional specification document for RIP (Routing Information Protocol), another for security features, and so on. Functional specifications are important because developers use them to create design documents. The documentation people also use them when they create manuals for end users.

1.1.3 Creating Architecture and Design Documents

When you have all of the requirements collected and arranged, it is the turn of the technical architecture team, consisting of highly qualified technical specialists, to create the architecture of the product. The architecture defines different components of the product and how they interact with each other. In many cases the architecture also defines the technologies used to build the product. While creating the architecture documents of the project, the team also needs to consider the timelines of the project. This refers to the target date when the product is required to be on the market. Many excellent products fail because they are either too early or late to market.

The marketing and sales people usually decide a suitable time frame to bring the product to market. Based on the timeline, the architecture team may drop some features of the product if it

is not possible to bring the full-featured product to market within the required time limits.

1.1.4 Implementation and Coding

The software developers take the design documents and development tools (editors, compilers, debuggers etc.) and start writing software. This is usually the longest phase in the product life cycle. Each developer has to write his/her own code and collaborate with other developers to make sure that different components can interoperate with each other. A revision control system such as CVS (Concurrent Versions System) is needed in this phase. There are a few other open source revision control systems as well as commercial options. The version control system provides

a central repository to store individual files. A typical software project may contain anywhere from hundreds to thousands of files. In large and complex projects, someone also needs to decide directory hierarchy so that files are stored in appropriate locations. During the development cycle, multiple persons may modify files. If everyone is not following the rules, this may easily break the whole compilation and building process. For example, duplicate definitions of the same variables may cause problems. Similarly, if included files are not written properly, you can easily cause the creation of loops. Other problems pop up when multiple files are included in

a single file with conflicting definitions of variables and functions.

1.1.5 Testing

Testing is probably the most important phase for long-term support as well as for the reputation of the company. If you don't control the quality of the software, it will not be able to compete with other products on the market. If software crashes at the customer site, your customer loses productivity as well money and you lose credibility. Sometimes these losses are huge. Unhappy customers will not buy your other products and will not refer other customers to you. You can avoid this situation by doing extensive testing. This testing is referred to as Quality

1.1.6 Software Releases

Before you start selling any software product, it is officially released. This means that you create a state of the software in your repository, make sure that it has been tested for functionality and freeze the code. A version number is assigned to released software. After releasing the software, development may continue but it will not make any change in the released software. The development is usually carried on in a new branch and it may contain new features of the product. The released software is updated only if a bug fixed version is released.

1.1.7 Documentation

There are three broad categories of documentation related to software development processes.

1. Technical documentation developed during the development process. This includes architecture, functional and design documents.
2. Technical documentation prepared for technical support staff. This includes technical manuals that support staff use to provide customer support.
3. End-user manuals and guides. This is the documentation for the end user to assist the user getting started with the product and using it.

All three types of documents are necessary for different aspects of product support. Technical documents are necessary for future development, bug fixes, and adding new features. Technical documentation for technical support staff contains information that is too complicated for the end user to understand and use. The support staff needs this information in addition to user manuals to better support customers. Finally each product must contain user manuals.

1.1.8 Support and New Features

Your customers need support when you start selling a product. This is true regardless of the size of the product, and even for products that are not software related. Most common support requests from customers are related to one of the following:

- The customer needs help in installation and getting started.
- The customer finds a bug and you need to release a patch or update to the whole product.
- The product does not fulfill customer requirements and a new feature is required by the customer.

In addition to that, you may also want to add new features to the product for the next release because competitor products have other features. Better support will increase your customer loyalty and will create referral business for you.