**KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**COLLEGE OF SCIENCE**

**DEPARTMENT OF COMPUTER SCIENCE**

**TITLE:**

**DESIGN AND IMPLEMENTATION OF A MULTI-LANGUAGE E-COMMERCE SITE**

**A Project Work Submitted in Partial Fulfilment of the Requirements For**

**BACHELOR OF SCIENCE IN COMPUTER SCIENCE**

**BY:**

**WILLIAM KPABITEY KWABLA (2565814)**

**SUPERVISOR:**

**MR. KWAKU AGYAPONG PABBI**

**April, 2018**

**©**

**CHAPTER THREE**

**PROJECT METHODOLOGY**

**3.0 Introduction**

In this chapter I take a look at the project methodology as well as the development tools that I will employ in developing this project. I will describe my selected methodology as well as the advantages and limitations of my selected methodology and also identify which online shop will be used as my case study.

* 1. **Development Methodology**

In software engineering, a system development methodology refers to the framework that is used to structure, plan, and control the process of developing an information system. Software development methodologies define the processes we use to build software. These methodologies are also referred to as Software Development Process Models. Each methodology follows a series of steps unique to its type to ensure success in the process of software development. A software process is a set of related activities that leads to the production of a software product. There are many different software processes but all must include four activities: specification, development, validation, and evolution that are fundamental to software engineering. A wide variety of Software Development methodology has evolved over the years, each with its own recognized strengths and weaknesses. The following are the most widely used methodologies for software development:

**3.1.1 Waterfall Method**

The waterfall model is a sequential approach, where each fundamental activity of a process represented as a separate phase, arranged in linear order. In the waterfall model, you must plan and schedule all of the activities before starting working on them (plan-driven process).

Plan-driven process is a process where all the activities are planned first, and the progress is measured against the plan. While the agile process, planning is incremental and it’s easier to change the process to reflect requirement changes.

**3.1.2 Incremental Method**

Incremental development is based on the idea of developing an initial implementation, exposing this to user feedback, and evolving it through several versions until an acceptable system has been developed.

**3.1.3 Spiral Method**

The spiral model is a risk-driven where the process is represented as spiral rather than a sequence of activities. It was designed to include the best features from the waterfall and prototyping models, and introduces a new component; risk-assessment. Each loop in the spiral represents a phase. Thus, the first loop might be concerned with system feasibility, the next loop might be concerned with the requirements definition, the next loop with system design, and so on.

**3.1.4 Prototyping Method**

A prototype is a version of a system or part of the system that’s developed quickly to check the customer’s requirements or feasibility of some design decisions. So, a prototype is useful when a customer or developer is not sure of the requirements, or of algorithms, efficiency, business rules, response time, etc.

The four basic process activities of specification, development, validation, and evolution

are organized differently in different development processes. In the waterfall model, they are organized in sequence, whereas in incremental development they are interleaved. How these activities are carried out depends on the type of software people, and organizational structures involved. In extreme programming, for example, specifications are written on cards. Tests are executable and developed before the program itself. Evolution may involve substantial system restructuring or refactoring.

For this project, I would engage with my users because their demands keep on changing, I will use the **Incremental method** to develop this application.

**3.2 Incremental Development Methodology**

Incremental development is based on the idea of developing an initial implementation, exposing this to user feedback, and evolving it through several versions until an acceptable system has been developed. In incremental development, the activities of a process are not separated but interleaved with feedback involved across those activities. An incremental method does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which is then reviewed to identify further requirements. This process is then repeated, producing a new version of the software at the end of each iteration of the method. During software development, more than one iteration of the software development cycle may be in progress at the same time. This process may be described as an "evolutionary acquisition" or "incremental build" approach. This means that the customer can evaluate the system at early stage in the development to see if it delivers what’s required. If not, then only the current increment has to be changed and, possibly, new functionality defined for later increments.

In this incremental method, the whole requirement is divided into various builds. During each iteration, the development module goes through the requirements, design, implementation and testing phases. Each subsequent release of the module adds function to the previous release. The process continues till the complete system is ready as per the requirement. The key to a successful use of an iterative software development lifecycle is rigorous validation of requirements, and verification & testing of each version of the software against those requirements within each cycle of the model. As the software evolves through successive cycles, tests must be repeated and extended to verify each version of the software.

Main principles of the Incremental development methodology are face-to-face meetings, constant cooperation, early and continuous delivery of the working software, transparency. Whenever there are unexpected or frequent changes either from the client’s side or internal, this model becomes the perfect choice for managers and team leaders.

In my case, the online shop is a web application and since user requirements may change with time, I will release the software as a series of increments to my supervisor, get his comments and suggestions and adjust the system and release subsequent versions until the final version is accepted by my supervisor (user). The whole process of specification, development and validation will undergo various iterations until the software is finally accepted by the end users (supervisor). Any recommendations or suggestions from the supervisor is factored into the next iteration until the supervisor accepts the final release.

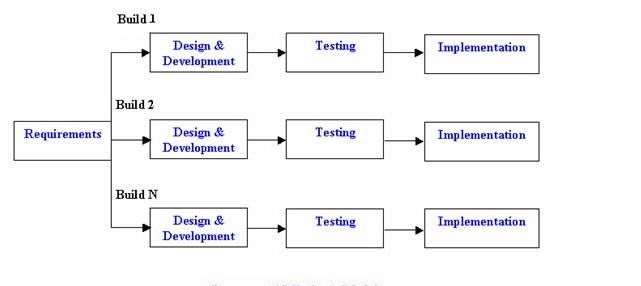


Figure 3.1 Incremental Development Methodology

However, there are advantages and limitations of the incremental development that we need to take a look at.

**3.2.1 Advantages** **of the Incremental Development Methodology**

* Risks are identified and resolved during iteration; and each iteration is an easily managed milestone**.**
* During the life cycle, software is produced early which facilitates customer evaluation and feedback**.** Because the first release is available to the user, they can give genuine and factual feedback based on the current system.
* Testing and debugging smaller modules and units within the application is much easier.
* It is possible to deliver applications more rapidly and quickly to the customer. Even if the full system functionality is not implemented in the initial release, users can use the system and enjoy all the benefits before a new version is released to cover all the functionalities.
* The cost of accommodating changing customer demands is reduced. The advantage is that it avoids or reduces rework. Because I engage the end user (supervisor) in the release of the increments or versions, the changing demand of the user (in this case the supervisor) is taken care of until I develop the full system.
* Better suited for large and mission-critical projects. This is because it is hard to break a small software system into further small serviceable increments/modules.
* Direct communication and constant feedback from customer (in this case the supervisor) leave no space for any guesswork in the system.

**3.2.2 Limitations of the Incremental Development Methodology**

* More resources may be required. This is because as new versions are released, I may bear additional cost as my user (in this case the supervisor) suggest new system functionalities to the next release.
* Needs a clear and complete definition of the whole system before it can be broken down and built incrementally.
* The project can get off track if my customer (in this case the supervisor) is not very clear about the final outcome of this project.
* System architecture or design issues may arise because not all requirements are gathered in the beginning of the entire life cycle.
  1. **Case Study Area / Project Case Study**

E-commerce websites in Ghana will be my target beneficiary for this project even though the application can be customized by any online shop to suit their operations. Ghana has a number of e-commerce websites including Jumia Ghana, Ahonya, Azaliabooks, and Zoobashop etc. Since my system would be for a single vendor, I chose Azaliabooks to be my case study. This is because my online shop would be used to sell Hardcopy Information Technology books and Azaliabooks also sells all kinds of e-books.

Azaliabooks.com is a website owned and operated by AZALIA. Their sole purpose is to promote the works of writers, publishers, photographers and other professionals in the creative arts industry, in order to explore, inspire and enrich lives. On their website, you can shop for electronic books (e-books) of all genres including educational materials, Christian literature, fiction & non-fiction novels, magazines, newspapers, etc. It also serves as an excellent online market for selling the soft copy of books.

* 1. **Development Tools**

In this project, a number of development tools would be used to complete this project.

* **PYTHON**

Is an interpreted, object-oriented, high-level programming language and a general-purpose programming with dynamic semantics and it will be used for the back-end.

It incorporates modules, exceptions, dynamic typing, very high level dynamic data types, and classes. Python combines remarkable power with very clear syntax. It has interfaces to many systems calls and libraries, as well as to various window systems, and is extensible in C or C++. It is also usable as an extension language for applications that need a programmable interface. Python is versatile. It runs websites and is used in many popular desktop applications on PCs and Macs. It can also be found in mobile applications and embedded in many devices. Python is also a popular scripting language for other applications. Python is portable, it runs on many Unix variants, on the Mac, and on PCs under MS-DOS, Windows, Windows NT, and OS/2.

* **DJANGO**

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Django is a free and open-source web framework which follows the model-view-template architectural pattern. Django is portable, it runs on many Unix variants, on the Mac, and on PCs under MS-DOS, Windows, Windows NT, and OS/2. The reasons why I chose Django for my project are:

1. **Fast:** Django has been designed in a way to help the developers make an application as fast as possible. From idea, production to release, Django helps in making it both cost effective and efficient. Thus, it becomes an ideal solution for developers having a primary focus on deadlines.
2. **Secure:** Django ensures that developers don’t commit any mistakes related to security. Some of the common mistakes include SQL injection, cross-site request forgery, clickjacking and cross-site scripting. To manage effectively usernames and passwords, the user authentication system is the key.
3. **Scalable:** To meet the heaviest traffic demand, the benefits of Django framework can be seen. Therefore, the busiest sites such as Instagram, Bitbucket, Pinterest etc. use this medium to quickly meet the traffic demands.
4. **Versatile:** Content management, scientific computing platforms, and even big organizations, all these aspects are very efficiently managed by the use of Django.
5. **Actively Developed:** One of the biggest risks of open source is whether there is sufficient interest in the project for it to attract developer support in the long term. There is no such risk with Django, not only is the project over 12 years old, it has a long history of consistent releases and it continues to be supported by an active community and a large core team of voluntary contributors who maintain and improve the code base every day. The Django development team maintains a development roadmap on the Django Project website and have a solid track record of meeting roadmap milestones. The Django Project is also supported by an independent foundation, the Django Software Foundation, that is a registered non-profit in the US.
6. **Stable Releases:** Open-source software projects are, in many cases, more actively developed and more secure than competing proprietary software. The downside of the ever-evolving development of an open-source software project is the lack of a stable codebase on which to base commercial development. Django addresses this issue with Long Term Support (LTS) versions of the software and a defined release process. LTS versions are released with a guaranteed (typically three years) support period. In this period the codebase is guaranteed to remain stable; with patches for bugs, security and data loss 100% compatible with the feature release. Django’s release process ensures that official releases are as stable as possible. After a development phase, each release enters an Alpha phase where a feature freeze is applied. The new release then moves through Beta and Release Candidate (RC) stages where bugs are worked out of the release. If no major bugs are found for a period after the release candidate, the final will be released (feature release). After the final has been released, only bugfixes and security patches are applied. These patches, like the LTS versions, are 100% compatible with the feature release.
7. **First Class Documentation:** A piece of software is only as good as its documentation. Django is very well documented and this allows avoiding of hours of trial and errors or effortless implementation. Every specific release of Django is supported by all necessary documentation and code examples. On top of that, the code is all publicly available on GitHub for direct investigation.

* **DJANGO REST FRAMEWORK**

It is a powerful and flexible toolkit for building Web APIs. The Django Rest Framework provides powerful model serialization, displays data using standard function-based views, or powerful class-based views for more complex functionality.

* **HTML**

HTML (Hypertext Markup Language) is the code that is used to structure a web page and its content. HTML is used to specify whether a web content should be recognized as a paragraph, list, heading, link, image, multimedia player, form, or one of many other available elements or even a new element that you define. It is the globally accepted programming language for formatting web pages. It is mostly used by small and medium scale businesses that do not really need advanced functionality on their websites. HTML is free, supports all browsers on the client’s machine, easy to use and understand hence, the choice in building the structure of my web pages.

* **CSS**

Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG or XHTML). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media. CSS is one of the core languages of the open web and is standardized across browsers according to the W3C (World Wide Web Consortium) specification.

* **JAVASCRIPT**

is a high-level, dynamic, weakly typed, prototype-based, multi-paradigm, and interpreted programming language. JavaScript is a full-fledged dynamic programming language that, when applied to an HTML document, can provide dynamic interactivity on websites. It would be used in conjunction with Django to ensure validation rules on the front-end of the websites.

* **BOOTSTRAP**

Bootstrap is a free and open-source front-end web framework for designing websites and web applications. It contains HTML and CSS based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only. Bootstrap would be used to design the styling of the application alongside CSS. Bootstrap is important in the application for the following reasons:

1. **Easy to use:** Anybody with just basic knowledge of HTML and CSS can start using Bootstrap.
2. **Responsive features:** Bootstrap's responsive CSS adjusts to phones, tablets, and desktops.
3. **Mobile-first approach:** Mobile-first styles are part of the core bootstrap framework.
4. **Browser compatibility:** Bootstrap is compatible with all modern browsers (Chrome, Firefox, Internet Explorer, Safari, and Opera).

* **PHOTOSHOP**

Adobe Photoshop is the predominant photo editing and manipulation software on the market. Its uses ranges from full featured editing of large batches of photos to creating intricate digital paintings and drawings that mimic those done by hand. It is a graphic designing tool that enables picture manipulation and editing. Photoshop would be used to design user interfaces and also the various images that would be required in developing the system.

* **POSTGRESQL** – It is an object-relational database management system (ORDBMS) with an emphasis on extensibility and standards compliance. A fundamental characteristic of an object-relational database is support for user-defined objects and their behaviors including data types, functions, operators, domains and indexes. This makes PostgreSQL extremely flexible and robust. Among other things, complex data structures can be created, stored and retrieved.

All the tables and records that would be required in the project would be designed and created with POSTGRESQL. My choice for POSTGRESQL is that it is easy to use, support is easily available on the internet, it is open source and hence inexpensive to get it, supports complex structures, it provides extensive data capacity and is trusted for its data integrity and finally it remains one of the most accepted industry standard database for developing web applications.

* **PYCHARM** – Pycharm is an Integrated Development Environment (IDE) used in computer programming, specifically for the Python language. It is the integrated development environment that would be use to write the executables of the Python and HTML files. I have chosen it because it allows developers to locate files very quickly and easily just a few key strokes, Multiple selections allows developers to interactively make changes to several lines of codes all at once, it allows developers to switch between several projects in the workspace, it has an integrated debugger. it is cross-platform and can be used on any operating system, and finally it allows key bindings, menus, snippets, macros and with Pycharm you need to spent little time to tune up your programming environment or hunt plugins for your basic development needs (Python, JavaScript, HTML, CSS). Pycharm does background spellchecking of written text and Python docstrings. It’s very handy for writing high quality software with meaningful comments and API descriptions. Pycharm has more robust integrated version control support (Git, SVN).
  1. **Summary**

This chapter begun with project methodologies where different methodologies were identified. I went further to describe software process model and I chose the incremental development process model under the agile development as the methodology to be used in this project. I identified the main activities of the incremental approach as *Requirements*, *Design and Development,* *Testing* and *Implementation* and also made it clear that the software would be released as a series of versions to my supervisor until the final product is accepted. Also, the advantages and disadvantages of the incremental development over other process models were analyzed. Furthermore, Azaliabooks was identified my case study and briefly description of their operations. Lastly, I analyzed the development tools used to develop my project. I delved much into the Django framework and discussed strong reasons why it is use it to develop my application as well as the other tools such as Python, PostgreSQL, HTML, CSS, Bootstrap, Adobe Photoshop, JavaScript Django Rest Framework and the Pycharm IDE.