

SOFTWARE ENGINEERING LAB REPORT 1

LIBRARY BOOK MANAGEMENT APP

PALAK SINGHAL	(16CO129)
SHARANYA KAMATH	(16CO140)

January 23, 2018

1. PROCESS MODEL

Spiral model

1.1 JUSTIFICATION

About Spiral Process Model:

In the spiral model, emphasis placed on risk analysis.

The spiral model has four phases:

- 1.1.1 Planning
- 1.1.2 Risk Analysis
- 1.1.3 Engineering
- 1.1.4 Evaluation

A software project repeatedly passes through these phases in iterations (called spirals).

The baseline spiral, starting in the planning phase, requirements are gathered and risk is assessed. Each subsequent spirals builds on the baseline spiral.

Why have we selected Spiral Model?

Following are the cases in which spiral model is used, and justification where these cases appear in our selected problem:

- *When costs and risk evaluation is important*
Our project doesn't have external funding, hence attention must be paid to cost. In spiral mode, risk evaluation and management is extremely efficient. This is an advantage for our software as risks and loopholes can be corrected easily.
- *Users are unsure of their needs*
We require our software to be flexible to change if more features are needed to be added in the later stages.
- *New product line*
This is our first software project so its definitely a new product line for us.
- *Significant changes are expected (research and exploration)*
If we come across risks which cannot be easily managed, we need to make some changes in our process planning.

Advantages of Spiral model:

- High amount of risk analysis hence, avoidance of risk is enhanced.
- Strong approval and documentation control.
- Additional Functionality can be added at a later date.
- Software is produced early in the software life cycle.

1.2 EXPLANATION

The steps / stages in the process model that would be suitable for our problem solving:

1. PLANNING PHASE

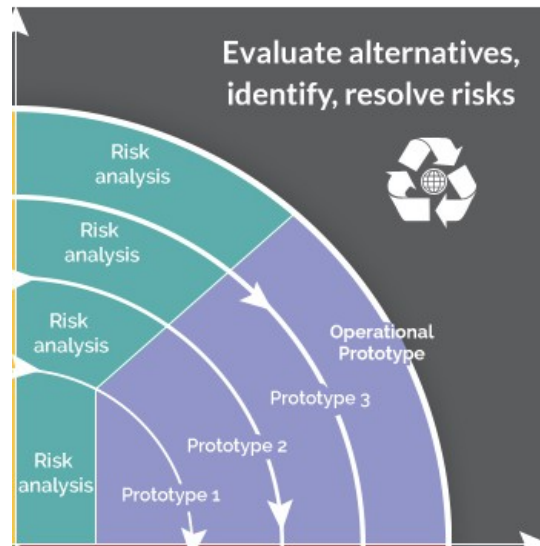
- Requirements for the software are gathered from the client.
- The feasibility of the software is studied and necessary changes are made.
- A documentation is created to understand the requirements.
- Identification of system, sub system and unit requirement is done.
- On accomplishing the spiral the project is deployed into the identified phase.



We as a part of our project will create a document which would state all the requirements for our software like the tools needed, the database required etc. We will also analyze the constraints and find some alternatives for the same.

2. RISK ANALYSIS:

- Brain storming sessions on the requirements are done to identify potential risks.
- After identification of risks they are prioritized according to their importance.
- Risk mitigation strategies are planned and finalized.
- Document highlighting all the risks is created.
- A prototype is produced at the end of this stage.

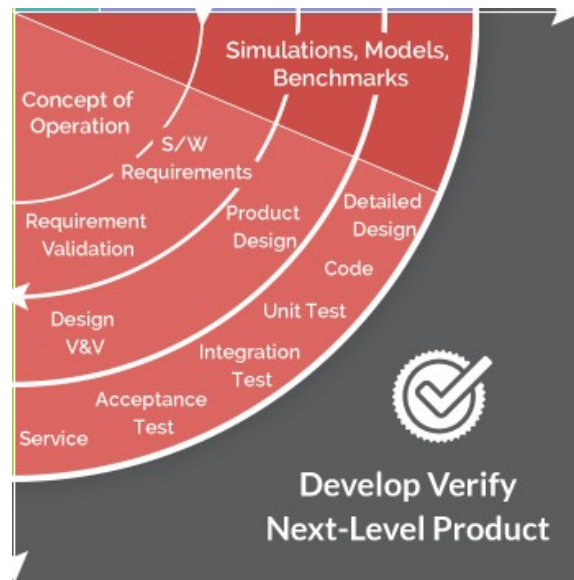


Our software might face a lot of risks at each phase.

Potential risks include loss of database due to hard disk corruption which could be resolved by backing up data. Other risks could be failure of a particular software tool and hence alternative tools should be kept ready as a part of risk management.

3. ENGINEERING:

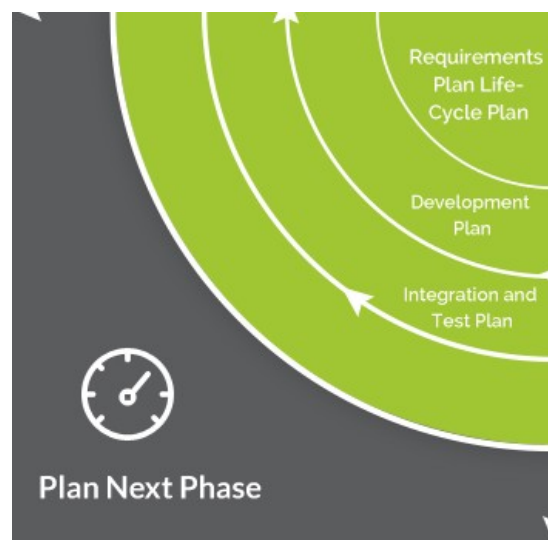
- Development and testing of the software is done.
- During initial spirals when the requirements are not so clear a proof of concept is made and sent for evaluation to the clients.
- In later spirals a detailed model called build could be created and sent to the clients for detailed feedback.
- Code for the software is written.
- Test cases are generated and the test results are reported.
- Test summary report and defect report is also made.



In this phase we will code our software according to the requirements specified in this spiral. The coding would be done using specific tools which will include the framework, front-end and back-end of the software. It will also be tested after that like testing whether the login bar works or not etc.

4. EVALUATION:

- Customers evaluate the software and give their feedback and approval.
- It is always done before going to the next spiral.
- A document is created stating all the features implemented so far.
- Next phase is also planned in this stage.



After each spiral we will evaluate our model and plan out the next phase.

2. TOOLS

Sr.No.	Step/Stage	Tool Name	About the Tool
1.	Requirement	OSRMT	Requirements management tool designed to achieve full SDLC traceability for features, requirements, design, implementation and testing. UI for requirements derivation, version control, attributes etc.
2.	Design	ARGOUML	ArgoUML is a free, open-source diagramming editor written in Java that will facilitate the job of creating diagrams that facilitate the planning and later implementation of applications or systems.
3.	Development	(a) Integrated Development Environment - INTELLIJ IDEA	The IDE provides certain features like code completion by analyzing the context, code navigation which allows jumping to a class or declaration in the code directly, code refactoring and options to fix inconsistencies via suggestions.
		(b) Application Development Frameworks, SDKs and APIs - ANDROID SDK	The Android SDK (software development kit) is a set of development tools used to develop applications for Android platform. The Android SDK includes the following: <ul style="list-style-type: none"> • Required libraries • Debugger • An emulator • Relevant documentation for the Android application program interfaces (APIs) • Sample source code • Tutorials for the Android OS

		(c) Database Tool - MONGO DB	MongoDB stores data in JSON-like documents that can vary in structure. Related information is stored together for fast query access through the MongoDB query language. This data model gives you the ability to represent hierarchical relationships, to store arrays, and other more complex structures easily.
		(d) Application Server - APACHE TOMCAT	Apache Tomcat is an open-source Java Servlet Container developed by the Apache Software Foundation (ASF). Tomcat implements several Java EE specifications including Java Servlet, JavaServer Pages (JSP), Java EL, and WebSocket, and provides a "pure Java" HTTP web server environment in which Java code can run.
		(e) Web Server - APACHE WEB SERVER	A Web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. Apache Web Server is an open-source web server creation, deployment and management software.
		(f) Build Tool - APACHE BUILDR	Buildr is an open source build system mainly intended to build Java applications. Buildr is based on Ruby's build system Rake, and uses Ruby as a scripting language.

4.	Testing	SELENIUM	Selenium is for automating web applications for testing purposes. It has the support of some of the largest browser vendors who have taken (or are taking) steps to make Selenium a native part of their browser. It is also the core technology in countless other browser automation tools, APIs and frameworks.
5.	Project Management	PROJECTLIBRE	ProjectLibre is an open source, freely available project management software system intended ultimately as a standalone replacement for Microsoft Project. ProjectLibre is written in the Java programming language, and will thus theoretically run on any machine for which a fully functioning Java Virtual Machine exists.

SUMMARY

In this report, the process model which is to be used for ***Library Book management app*** is identified and a brief description is given of the same. The spiral model is chosen, which has 4 phases starting with planning, risk analysis , engineering and ending with evaluation. The reasons for choosing the spiral model and suitable justifications are also provided.

Advantages of the spiral process model are elaborated.

The steps/stages in the process model for the above problem are explained in detail.

All the tools which are required during the implementation of various phases are stated, like requirements tools for the requirement elicitation phase, design tools for the design phase etc.

This report helps in understanding the project in depth by giving an idea on how to go about it, the process model to be used and also gives an insight on the various phases in software development cycle. Introduction and description is given of various open source tools that could be used during the various stages of the project.