DEPARTMENT OF
COMPUTER SCIENCE AND ENGINEERING

Lab No. 05
Title: SDLC model selection for the given project.

INTEGRATED DESIGN PROJECT I

CSE 324

GREEN UNIVERSITY OF BANGLADESH

1 Objective(s)

- The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

2 Problem analysis

See the problem details in lab manual 01.

3 Methodology

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process. A typical Software Development Life Cycle consists of the following stages −

3.1 Stage 1: Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas. Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

3.2 Stage 2: Defining Requirements

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an SRS (Software Requirement Specification) document which consists of all the product requirements to be designed and developed during the project life cycle.

3.3 Stage 3: Designing the Product Architecture

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification. This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product. A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

3.4 Stage 4: Building or Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle. Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

3.5 Stage 5: Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where

product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

3.6 Stage 6: Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing). Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

4 SDLC Models

Following are the most important and popular SDLC models followed in the industry –

- Waterfall Model

- Iterative Model

- Spiral Model

- V-Model

4.1 Waterfall Model

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model, phases do not overlap.

This model is used only when the requirements are very well known, clear and fixed.
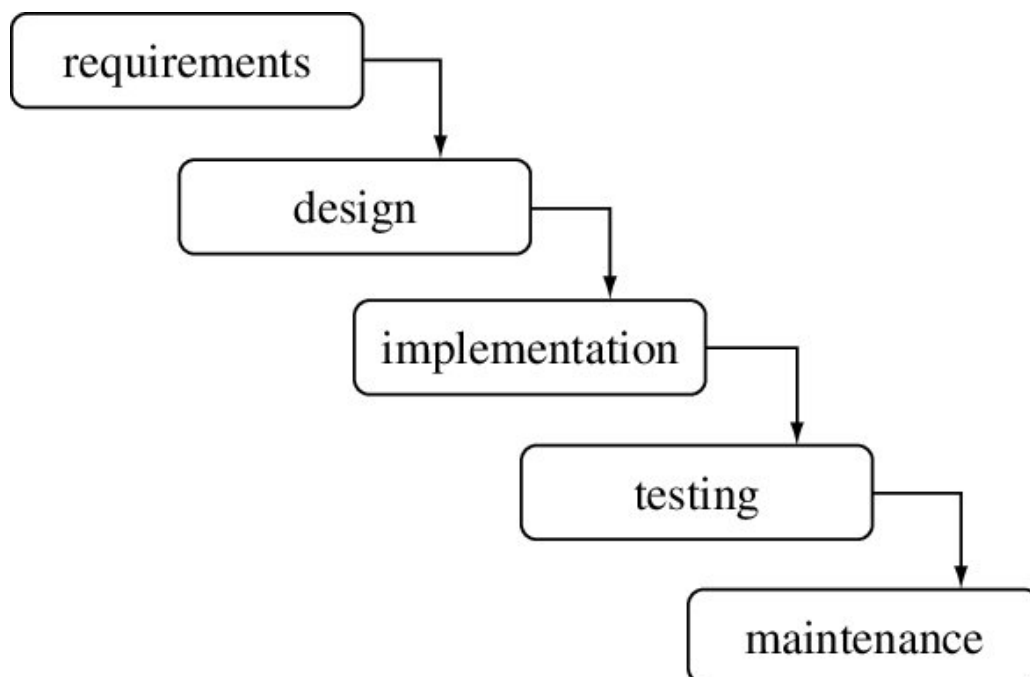

Figure 1: Waterfall model

4.2   Iterative Model

Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

When requirements are defined clearly and easy to understand, When the software application is large and When there is a requirement of changes in future, this model is used.
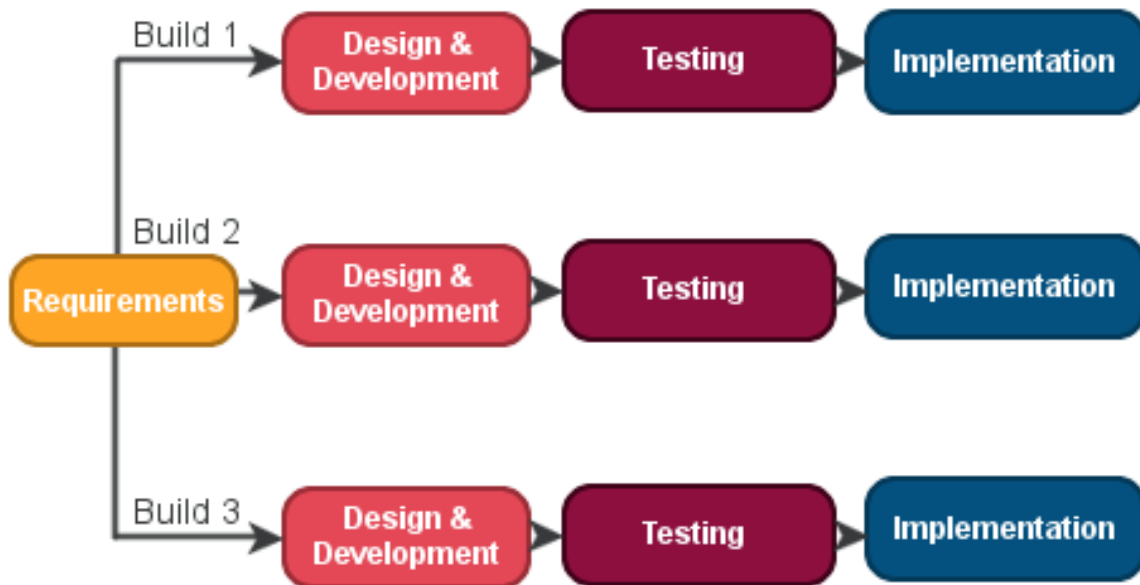

Figure 2: Iterative Model

4.3   V-Model

Under the V-Model, the corresponding testing phase of the development phase is planned in parallel. So, there are Verification phases on one side of the 'V' and Validation phases on the other side. The Coding Phase joins the two sides of the V-Model.

V-Model is used for small projects where project requirements are clear, Simple and easy to understand and use. This model focuses on verification and validation activities early in the life cycle thereby enhancing the probability of building an error-free and good quality product
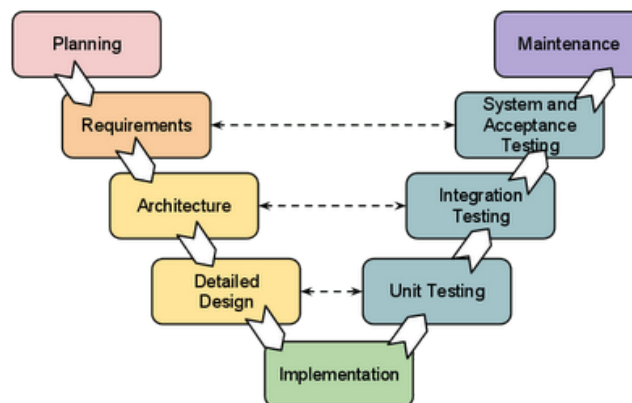

Figure 3: V-shape Model

4.4 Spiral Model

The spiral model has four phases – Identification, design, build, evaluation and risk analysis. A software project repeatedly passes through these phases in iterations called Spirals.

A Spiral model in software engineering is used when project is large, When releases are required to be frequent, When creation of a prototype is applicable and When risk and costs evaluation is important.
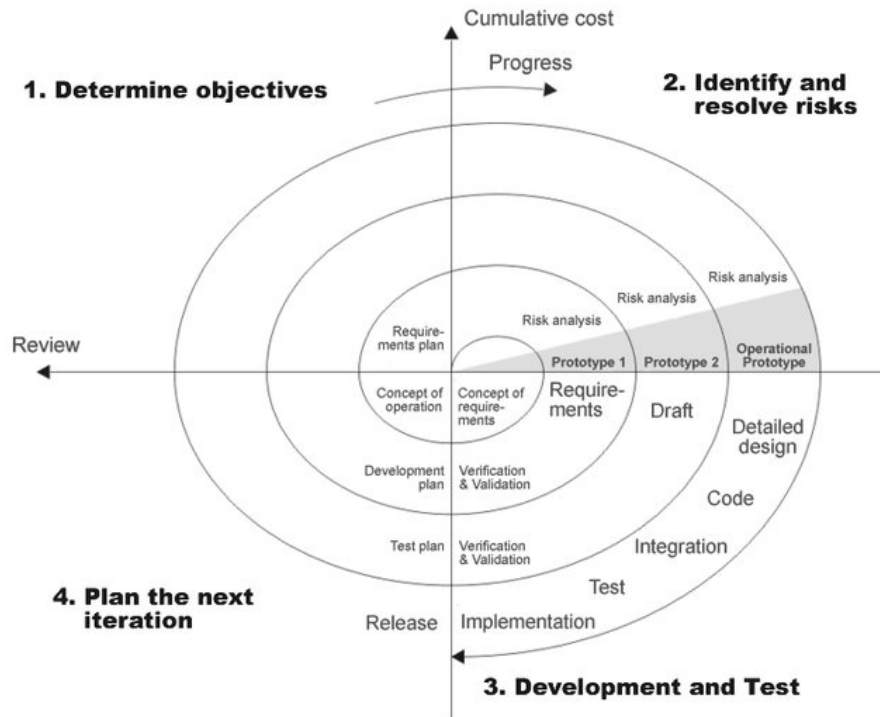

Figure 4: Spiral Model

4.5 Agile Model

The meaning of Agile is swift or versatile."Agile process model" refers to a software development approach based on iterative development. Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning.

The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.

4.6 Prototyping Model

Prototyping is defined as the process of developing a working replication of a product or system that has to be engineered. It offers a small scale facsimile of the end product and is used for obtaining customer feedback.
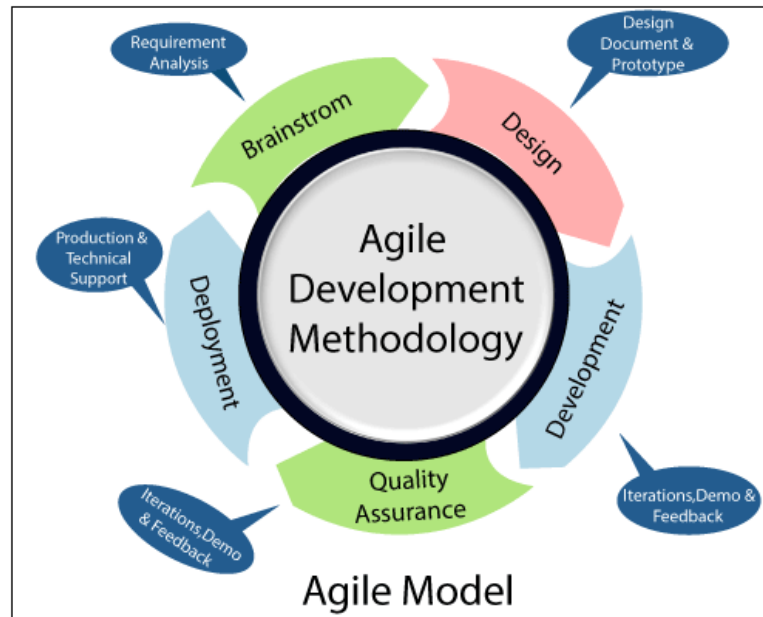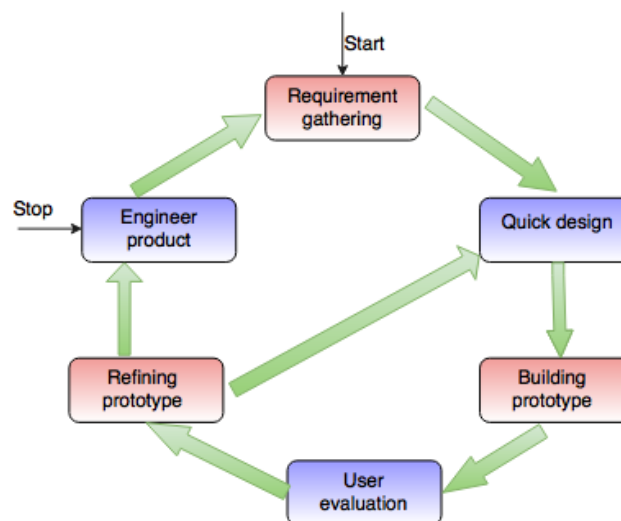
Figure 5: Agile Model

Figure 6: Prototype Model

5 Implementation in ATM Management system

Here we can see a table for SDLC Model for our project. We chose some category which is suitable for our project. Then we select some priority above the requirement for our project. Then see which category is support by which model. Then we chose the best suitable Model above the the priority we set for each category. Then we will set this SDLC Model for our Project.

In this SDLC Models, Waterfall scores 11, V-Shaped scores 23, Iterative scores 11, Spiral scores 22, Agile scores 5 and Prototype scores 11. Here our most important categories are supported by V-Shaped Model. So we will prefer V-Shaped Model as a best option for our project.

Table 1: Comparison matrix with different models

Priority	Criteria	Waterfall	V-shape	Iterative	Spiral	Agile	Prototype
5	Well known requirement	Yes	Yes	No	No	No	No
3	Technological knowledge	Yes	Yes	No	No	No	No
6	Efficiency	No	Yes	Yes	Yes	No	Yes
3	Risk analysis	No	No	No	Yes	No	No
5	User testing ability	No	No	Yes	Yes	Yes	Yes
5	Dependability and Security	No	Yes	No	Yes	No	No
3	Time consuming	Yes	Yes	No	Yes	No	No
Total- 30	Over all	11	23	11	22	5	11

6 Discussion & Conclusion

Based on the focused objective(s), to understand about the right choice of model selection, the additional lab exercise made us more confident towards the fulfilment of the objectives(s).

7 Lab Task (Please implement yourself and show the output to the instructor)

1. Select the best model of a library management system by creating a model matrix.

8 Lab Exercise (Submit as a report)

- Think about the best model for your selected system with your partners and submit a report touching the above criteria.

9 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.