

SDLC

Model

Report

Members:

120CS0673- Shaon Kumer Paul

120CS0674- Kodidela Sanjay Reddy

120CS0675- Medam Indra Sena Reddy

120CS0676- Nenavath Mahendra Sai

Analysis:

Features	Analysis
1. Requirement Specification	Beginning
2. Understanding requirements	Well Understood
3. Cost	Low
4. Availability of reusable components	No
5. Complexity of system	Simple
6. Risk Analysis	No Risk Analysis
7. User involvement in all phases of SDLC	Intermediate
8. Guarantee of Success	Good
9. Overlapping Phases	Less
10. Implementation time	Less
11. Flexibility	Flexible
12. Changes Incorporated	Easy
13. Expertise Required	Medium
14. Cost Control	No
15. Resource Control	No

Assuming that the development team has experience with similar projects it's safe to choose Waterfall Model for development of the project.

Classical Waterfall Model

1. Feasibility study:

➤ Financial Feasibility

The cost and benefit of the project is analysed. Means under this feasibility study a detailed analysis is carried out of what will be the cost of the project for development which includes all required cost for final development like hardware(monitors, hardware, external storage, keyboards, mouse, printers etc) and software resource(database services etc) required, design and development cost and operational cost and so on. After that it is analysed whether the project will be beneficial in terms of finance for the organisation or not. Cost estimation is performed using Cocommo model

➤ Technical Feasibility

In Technical Feasibility current resources both hardware software along with required technology are analysed/assessed to develop Book-shop automation software(BAS). This technical feasibility study reports that there exists correct required resources and technologies which will be used for BAS development. Along with this, the feasibility study also analyses technical skills and capabilities of the technical team, whether existing technology can be used or not, maintenance and up-gradation is easy or not for chosen technology etc.

➤ Schedule Feasibility

In Schedule Feasibility Study, timelines/deadlines are analysed for BAS which includes how many times teams will take to complete the final project which has a great impact on the organisation as the purpose of the project may fail if it can't be completed on time.

We can conclude that BAS is feasible to do and generate a feasibility report.

2. Requirements Analysis:

➤ Eliciting Requirements

○ Step 1: Identify the stakeholders

1. Anyone who operates the system
2. Anyone who benefits from the system
3. Anyone involved in purchasing or procuring the system.
4. Organisations which regulate aspects of the system
5. People or organisations opposed to the system

Based on this we can classify the users as:

- a. Key Stakeholder: Book-shop Owner, Manager
- b. Primary Stakeholder: Employees, Sales Clerk
- c. Tertiary Stakeholder: Customers

○ Step 2: Requirement Gathering

- a. Meetings with project managers, stakeholders and users are held in order to determine the requirements like; who is going to use the system? How will they use the system? What data should be input into the system? What data should be output by the system? These are general questions that get answered during a requirements gathering phase.

- b. Additionally,

A formal interview with the Book-shop Owner, Manager, Employee, Clerk is sufficient to know all the problems faced and the individual requirements/functionalities for each user.

Questions asked in the interview of around 10-15 min:

- i. Problems facing in manual process
- ii. Additional features
- iii. Expectation of the software based on representation and performance
- iv. Statistics on No. of purchases, books, employees etc. (For scaling purpose)

➤ Analysing Requirements

Based on the requirements provided by the stakeholders we can determine that the stated requirements are clear,

complete, consistent and unambiguous. Requirements are analysed for their validity and the possibility of incorporating the requirements in the system to be developed is also studied.

➤ **Recording Requirements**

Finally, a Requirement Specification document is created which serves the purpose of guideline for the next phase of the model. Requirements are documented in various forms, including a summary list and include natural-language documents, use cases, user stories, process specifications and a variety of models including data models

3. Design:

➤ **Logical Design**

- Abstract representation of the data flows, inputs and outputs of the system.
- This is often conducted via modelling
- Logical design includes creation of entity-relationship diagrams(ER diagrams).

➤ **Physical design**

- In physical design, the input, output, storage, processing, and recovery requirements about the system are decided.
- User Interface Design
- Data Design: concerned with how the data is represented and stored within the system.
- Process Design: concerned with how data moves through the system (DFD)

➤ **Architectural design**

- Design of the system architecture that describes the structure, behaviour and more views of that system and analysis.

Creation of DFD, UML diagrams etc to define the overall system architecture. The system design specifications serve as input for the next phase of the model.

4. Coding and Integration:

On receiving system design documents, the work is divided in modules/units and actual coding is started. Since, in this phase the code is produced so it is the main focus for the developer.

Implementation of the

1. Front-end:

Website done using HTML, CSS, JavaScript

2. Back-end:

Flask, Json, Python

Integrate both the frontend and backend for hosting the website

5. Testing and Debugging:

Perform unit testing, integration testing, system testing to identify the faults and use Debugging to find the source of already identified defect and to fix it.

Steps in debugging:

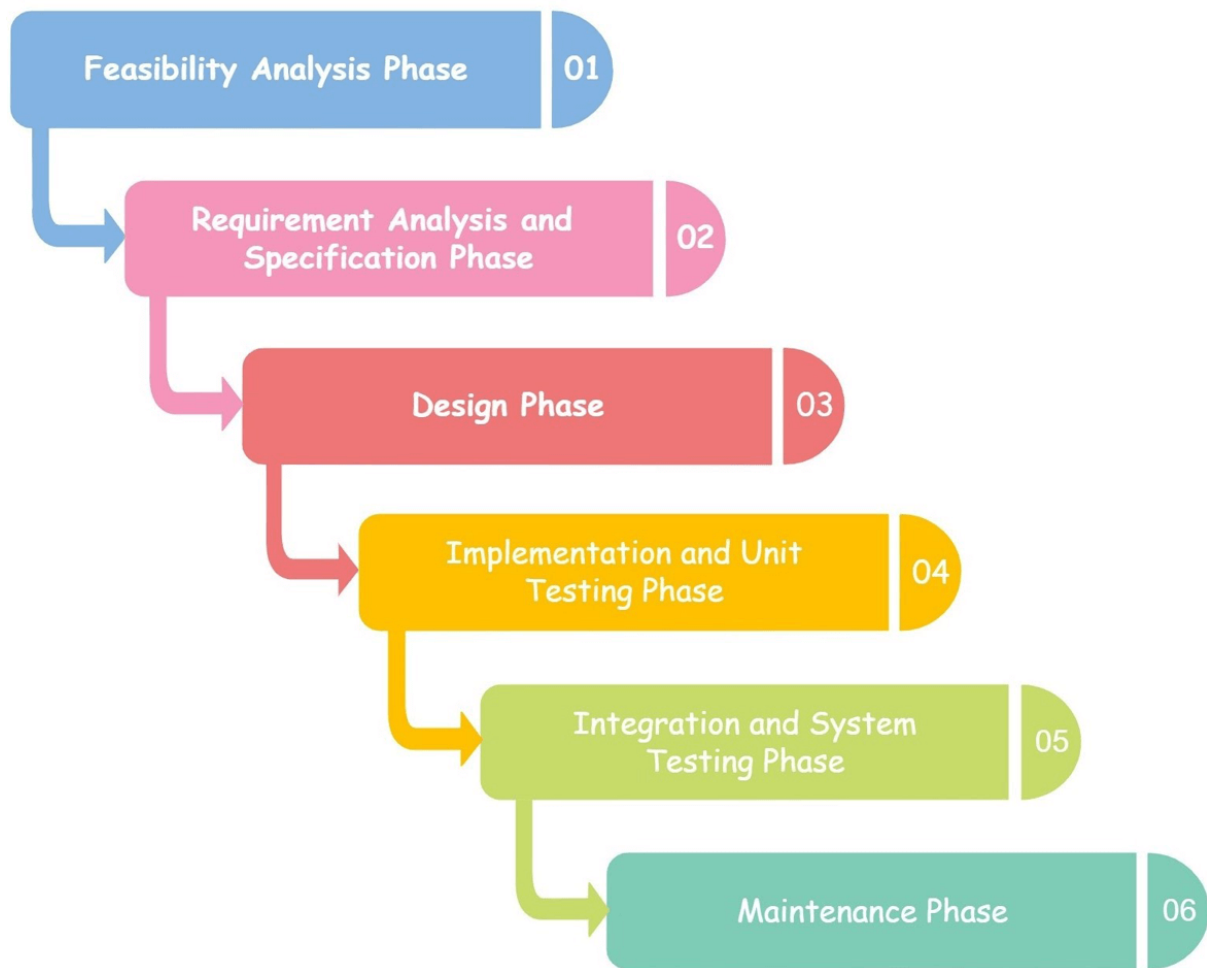
- Attempt to reproduce the problem.
- After the bug is reproduced, the input of the program may need to be simplified to make it easier to debug.
- After the test case is sufficiently simplified, a programmer can use a debugger tool to examine program states (values of variables, plus the call stack) and track down the origin of the problem(s).
- Fix the defect
- Test to check if the fix is correct

6. Implementation and Maintenance:

Deployment activities

- Release
- Installation and activation
- Deactivation
- Uninstallation
- Update
- Version tracking

Once when the customers start using the developed system then the actual problems come up and need to be solved from time to time.



Finally the Bookshop Automation software is developed and is made available in use by the customers.

Precautions:

1. As we are using the waterfall model, Integration is one big bang at the end. So development phase must be performed with utmost care

Advantages:

1. Easy to understand, easy to use, especially by inexperienced staff
2. Milestones are well understood by the team
3. Provides requirements stability during development
4. Facilitates strong management control (plan, staff, track)