EXPERIMENT NO: 02

Aim: Identify the objectives and summarize outcomes for given scenario, for each SDLC phase.

Theory:

- **Software Development Life Cycle:**

  A software development process life cycle (SDLC) is a framework defining tasks performed at each step in the software development process. SDLC is a structure followed by a development team within the software organization. It consists of a detailed plan describing how to develop, maintain and replace specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

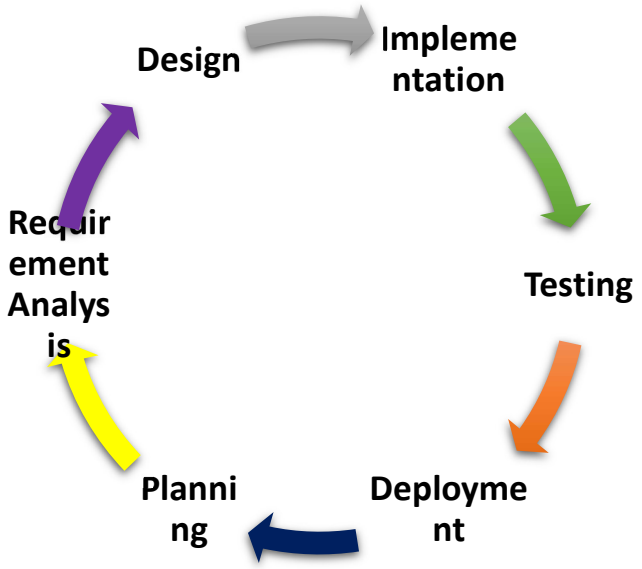

Fig: SDLC phases

- **SDLC Phases**

  1. Planning
  2. Requirement Analysis
  3. Design
  4. Implementation

5. Testing
6. Deployment and Maintenance

- **Planning**

The first step in the SDLC defines the scope of the project. For example, your sales reps might be stuck with a hodgepodge of apps on Androids and iPhones to track sales leads. You, the owner, want them on a unified system that works better with your company's internal software. This may mean changing to a single type of hardware and choosing sales tracking software managed by the home office. System planning is the process of deciding what your new information system should look like and then identifying the resources needed to develop it.

- **Requirement Analysis:**

1. **Communication**

Another Part of Requirement Analysis is Communication where the stakeholders discuss the requirements of the software that needs to be developed to achieve a goal. The aim of the requirement analysis phase is to capture the detail of each requirement and to make sure everyone understands the scope of the work and how, spiral development and each requirement is going to be fulfilled

2. **Requirement Gathering**

Requirements Gathering (also known as Requirements elicitation or Capture) is the process of generating a list of requirements (functional, system, technical, etc.) from the various stakeholders (customers, users, vendors, IT staff, etc.) that will be used as the basis for the formal Requirement Definition. The process is not as straightforward as just asking the stakeholders what they want they system to do, as in many cases, they are not aware of all the possibilities that exist, and may be limited by their immersion in the current.

3. **Defining Requirement**

Ones the requirement analysis is doe next step is to delay defining the documents the product requirement and get them approved from the customer or the market analysts. This is done

through an SRS document which is consist of all the product requirement to be designed and deployed during the project life cycle.

It Also Consist: 1. Cost Estimation

- **Design and Functionality:**

  Functionality

  Step three in the SDLC is reserved for listing features that support the system's proper functioning. For example, an inventory control system may need to handle at least 15 users or that it should interface with the U.S. Customs database as a compliance check on imports.

  1.Tracking

  2.Scheduling

  Design

  1. Algorithm
  2. Flowchart

- During the design phase, developers and technical architects start the high-level design of the software and system to be able to deliver each requirement.

- The technical details of the design are discussed with the stakeholders and various parameters such as risks, technologies to be used, capability of the team, project constraints, time and budget are reviewed and then the best design approach is selected for the product.

- The selected architectural design, defines all the components that needs to be developed, communications with third party services, user flows and database communications as well as front-end representations and behavior of each components. The design is usually kept in the Design Specification Document (DSD)

- **Implementation/Coding**

  After the requirements and design activity is completed, the next phase of the Software Development Life Cycle is the implementation or development of the software. In this phase, developers start coding according to the requirements and the design discussed in previous phases. Database admits

create the necessary data in the database, front-end developers create the necessary interfaces and GUI to interact with the back-end all based on guidelines and procedures defined by the company.

- **Testing**

After the code is developed it is tested against the requirements to make sure that the product is actually solving the needs addressed and gathered during the requirements phase. During this phase all types of functional testing like unit testing, integration testing, system testing, acceptance testing are done as well as non-functional testing are also done.

This cycle is repeated until all requirements have been tested and all the defects have been fixed and the software is ready to be shipped.

- **Deployment and Maintenance**

Once the software has been fully tested and no high priority issues remain in the software, it is time to deploy to production where customers can use the system. Once a version of the software is released to production, there is usually a maintenance team that look after any post-production issues .If an issue is encountered in the production the development team is informed and depending on how severe the issue is, it might either require a hot-fix which is created and shipped in a short period of time or if not very severe, it can wait until the next version of the software.

- **Scenario: "Development of employment salary attendance and biodata management system for a huge government company"**

I. Planning

- Firstly, we discussed the various features and functions of the software to be developed such as attendance report generation per employee/users, editing user details and other functions.
- We also planned which framework to use for the software application while planning as well as which parts can be reused or open source codes.
- Along with that e also planned about the hardware to be used in the development of the application, we planned to have a user log in through the company's website using his credentials due to the work from home situation, and otherwise we planned to install a fingerprint / ID-card scanner in the office to check-in and check-out incase for the office jobs.
- All the work to be done was planned and organized based on the capabilities of the team members.

II. Requirement Analysis

- The foremost part of requirement analysis is communication. This is where the stakeholders discuss the requirements of the software that needs to be developed to achieve a goal.
- In this phase we communicated with the stakeholders to get an overall understanding of the requirements of the projects and to capture its details and also making sure that everyone understands the scope of the work was also looked after during this process.
- Once the requirement analysis was done, the SRS document of the project was designed after the approval from the stakeholders and was deployed. It also consists of the cost estimation.

III. Modelling

- In the modeling phase of the Software Development Life Cycle, models/prototypes of the software to be created are made in order to better understand the requirements of the software and the design that will achieve these requirements.
- In our software project the modelling activity consisted of two phases namely, analysis and designing.
-

i) Analysis:

In analysis phase, we analyze the need and working of every model. There are three types of models in our project, namely, the GUI (Graphic User Interface) of the application, the flow model of the software application and the database relationship model of the application's database. The analysis of these models to be made will be done in this phase. Further in the design phase the models will be made.

ii) Designing:

In designing phase, the models/prototypes for the software application will be made from the details confirmed from the analysis phase in the modelling.

IV. Construction

- The construction phase is mainly divided into two parts, code generation and software testing.
- In the code generation phase, the software application is coded by the developers.
- For our software project, 'Employment salary attendance and biodata management system', we used flutter and dart for making the android applications and firebase with the use of NOSQL as the database for our application.
- For the web version of the software application, we used HTML, CSS, JavaScript and PHP as backend language along with firebase as database.
- In the testing phase, the software is given to the testers to find the vulnerabilities of the software and to catch any bugs missed by the developers and the priority is finding the bugs fast and getting them fixed.

- After the testing is done and no bugs that can cause an adverse effect on the software application are found then the software product is set for the deployment stage.

V. Deployment

- The software application is given to the customer who evaluate the delivered product and provides feedback based on the evaluation. The feedback that the customer will gives will be used to determine the future additions/updates to be made to the software.

• Questions:

1. Which process according to you works simultaneously and Why?

According to me, the processes communication and modelling are the phases that can work simultaneously. As while communicating and forming ideas in our mind we can create a rough model of the software to be made by getting the specific requirements and details from the customers, and as we get these details we can simultaneously make diagrams and other documents for our modeling phase.

2. Which of the following is the most important phase in SDLC? why?

- I think that all the phases of the SDLC have their own importance but the communication phase is the most important phase of the SDLC according to me.
- The communication phase is the phase where all the stakeholders of the project communicate regarding the software project's details, specific requirements required by the customers as well as the team members.
- Also, the majority of the bugs in the software applications (around 55%) happen due to lack of communication between the stakeholders. As if there are any miscommunication between the stake holders or the team members.
- There is also a risk of failure if there is lack of communication between the stakeholders as it might result into unclear specification and creation of a wrong software product.
- For this reason I think that the communication phase is very crucial and according to me is the most important phase of the Software Developmental Life Cycle.

3. From where do defects and failures in software testing arise?

- The defects and the failures in the process of software testing arises from the errors in the communication, modeling and construction phases.

- The lack of communication or miscommunication in the communication phase can result in errors in the software specifications ultimately resulting errors in the software application.
- The errors in modelling phase can include UI related errors. For example the screen not being responsive, hard to understand the UI. It can also include the database related errors and software model related errors.
- Whereas the errors in the construction phase include run time errors and logical errors which might be cause by the developer's mistake.
- The other types of errors that occur are errors while using the system, intentionally caused damage, environmental conditions (specifications of the device),etc.

4. Which phase of the SDLC is known as the "ongoing phase" and why?

- The maintenance phase of the SDLC is known as the "Ongoing phase".
- Maintenance is the only phase which is in action after the product has been delivered to the customer.
- As our product is a software application, it will need changes/updates according to the customer's needs.
- The maintenance phase involves making changes to the software according to the changes requested by the customer, enhancing security, the bug fixes missed by the testing team, etc.
- Thus, Maintenance phase is the phase which is also known as 'Ongoing phase' in the Software Development Life Cycle.

- **Conclusion:**

Hence, by performing this practical I learnt about the Software Development Life Cycle (SDLC), the various phases of SDLC, and I also identified the various objectives and summarized the outcomes for the 'Employment salary attendance and biodata management system for a huge government company'.

(10)	(20)	(10)	(10)	Total (50)