**Django Web Application:**

**Initial Requirements:**

1. Develop a application with authentication system
2. Home to be like a dashboard to
   1. show employee performance information
   2. show work anniversary
   3. birthdays
   4. company news information
   5. insurance information
   6. and some more
3. Develop navigation bar to go to different features
   1. Apply timesheet
   2. Apply leaves
   3. Approve leaves
   4. Approve timesheet
   5. Employee search database
   6. Look for holidays
   7. Look for company policies
   8. Salary information
   9. Profile information

**Accepted Requirements**

1. Develop a login application
   1. Develop a form to intake username -> employee id -> only digits
   2. Password box and a login button
   3. A link to new registrations
      1. Username
      2. Email address
      3. Employee id
      4. Password
      5. Confirm password
   4. url to be /auth/login/
   5. redirect to home page after login /auth/home/
2. Develop a application to store employee profile information
   1. Name, employee id
   2. Date of birth – must be a calender
   3. Profile pic – upload file
   4. Date of joining – must be a calender
   5. Primary skills – must be a dropdown
   6. Manager – must be dropdown
   7. Project – must be a dropdown
   8. You may add some more
   9. Model name - Profile
3. Holiday information
   1. Provide a link in navigation bar
   2. When clicked on link – show list of holidays for current year
   3. Model name – Holidays
4. Develop a employee search database
   1. An employee can be searched by following ways
      1. Employee Number
      2. Employee Name
      3. By Manager Employee ID
   2. List of all employees to be shown in a list format
   3. If clicked on listed employee name then details of employee must be shown
5. Home Dashboard
   1. url information
   2. Create panels to show dashboard information
      1. Work anniversary -> get all employees joining date and see if it falls in the current week and show them on dashboard
      2. Provide a link on every username, when clicked must show details of employees
      3. Birthdays -> similar to above
      4. Apply leave application
         1. Create a form to select 2 date ranges ie start date and end date and a submit button
         2. Start date must be less than end date
         3. Given a submit button to submit the leave
         4. Upon submit a email notification must be sent to manager
         5. Once manager approves leave notification must be sent to employee who has applied leave
         6. If leaves are applied on public holidays or on weekends, application should take care of these
         7. Application must support to apply sick leave, earned leave, paid leave etc category
         8. If no sufficient leaves are available then application must error
         9. In the top of the page leave information must be displayed stating -> every type of leave with balance and used leave information
         10. Model name to keep leave balance information
         11. Model name to keep track of leaves applied
      5. Apply timesheet
         1. Create a form to select 2 date ranges ie start date and end date and a submit button
         2. Upon submitting a form must be displayed which is prefilled with employees project information and elaborated date
         3. Employee can fill no. of hours worked and the description to submit the timesheet for selected date range
         4. Notification will be sent to manager after timesheet submission
         5. Notification will be sent to employee after manager accepts/rejects the timesheet
         6. Employees should not be allowed apply timesheet on weekends, public holidays and on the days when employee was absent
         7. Model name to keep track of timesheet data

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality softwares. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.

* SDLC is the acronym of Software Development Life Cycle.
* It is also called as Software Development Process.
* SDLC is a framework defining tasks performed at each step in the software development process.
* ISO/IEC 12207 is an international standard for software life-cycle processes. It aims to be the standard that defines all the tasks required for developing and maintaining software.

What is SDLC?

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.

The following figure is a graphical representation of the various stages of a typical SDLC.



A typical Software Development Life Cycle consists of the following stages −

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.

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.

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.

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.

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.

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.