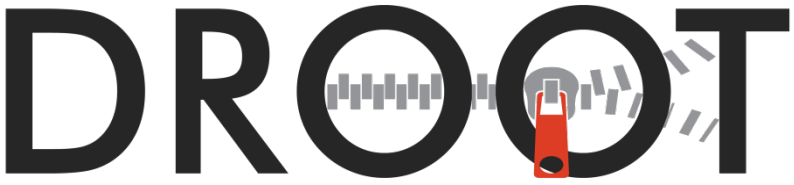


YOUR PERSONAL TRAVEL ASSISTANT

DROOT

TOPIC – Personal Travel Assistant.

CATEGORY- Dynamic web-based application.

Problem Statement

- Keep forgetting important stuff before travelling
- Sick of keeping all documents in an unorganized way
- Keeping live updates of your travel companions
- Don’t know what other companions have packed

Solution

- Building a platform that will assist the user before and during the trip.
- Connecting all the companions thus keeping them updated.
- Creating an online marketplace for all travel related activities
- Developing mobile application to provide a beautiful and intuitive interface, making it easy to access and operate for any age group.

DROOT

Layers -

Tools –

Operating Systems	-Windows
Programming Language	-Flutter, Dart
Database	-Firebase
Server	-Apache tomcat / Glassfish server
IDE	-Net beans, Android Studio

Method-

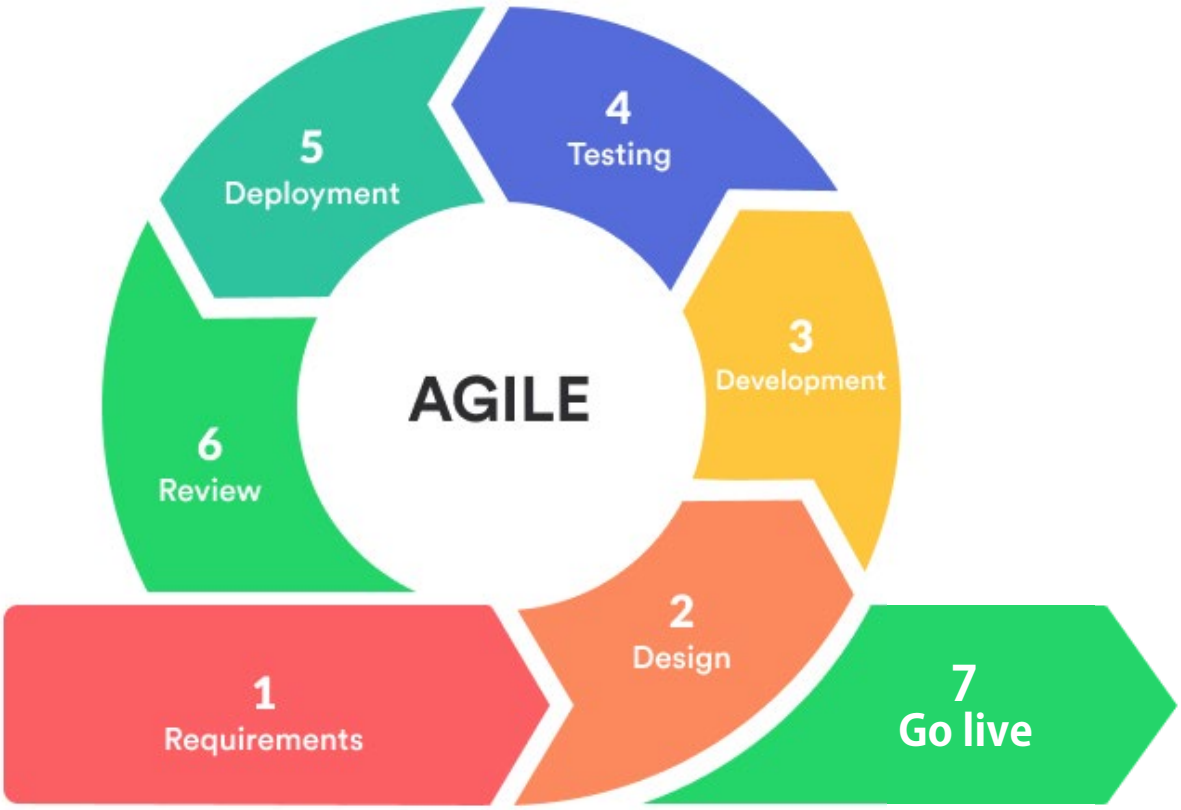
This application works on a dynamic web based system which acts as a personal travel assistant and will assist the user before, during and after the trip.

Model-

In this application we will use **Agile SDLC model**.

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds.

DROOT


AGILE
1
Requirements
2
Design
3
Development
4
Testing
5
Deployment
6
Review
7
Go live

Comparison of different life cycle models

Classical Waterfall Model-

The Classical Waterfall model can be considered as the basic model and all other life cycle models are based on this model. It is an ideal model. However, the Classical Waterfall model cannot be used in practical project development, since this model does not support any mechanism to correct the errors that are committed during any of the phases but detected at a later phase. This problem is overcome by the Iterative Waterfall model through the inclusion of feedback paths.

Iterative Waterfall Model-

The Iterative Waterfall model is probably the most used software development model. This model is simple to use and understand. But this model is suitable only for well-understood problems and is not suitable for the development of very large projects and projects that suffer from a large number of risks.

Prototyping Model-

The Prototyping model is suitable for projects, which either the customer requirements or the technical solutions are not well understood. These risks must be identified before the project starts. This model is especially popular for the development of the user interface part of the project.

Evolutionary Model-

The Evolutionary model is suitable for large projects which can be decomposed into a set of modules for incremental development and delivery. This model is widely used in object-oriented development projects. This model is only used if incremental delivery of the system is acceptable to the customer.

Spiral Model-

The Spiral model is considered as a meta-model as it includes all other life cycle models. Flexibility and risk handling are the main characteristics of this model. The spiral model is suitable for the development of technically challenging and large software that is prone to various risks that are difficult to anticipate at the start of the project. But this model is very much complex than the other models.

Agile Model-

The Agile model was designed to incorporate change requests quickly. In this model, requirements are decomposed into small parts that can be incrementally developed. But the main principle of the Agile model is to deliver an increment to the customer after each Time-box. The end date of iteration is fixed, it can't be extended. This agility is achieved by removing unnecessary activities that waste time and effort.