# Background

This chapter discusses the key concepts that are required to understand the thesis work.

## Static Analysis

In the last few years, we see the enormous increase in the usage of software. We see software everywhere in the walk of our life with the Internet of Things, for example. As the usage increases, it stresses quality as being more secure and does not break. Earlier, software developers used to do manual auditing of the code, but sooner they realised it is time consuming and so planned to automate the process. Therefore, different testing mechanisms evolved such as black-box testing, white-box testing.  
Static analysis falls under the category White-box testing. It tests the software without executing the code, and therefore the name means it. On the other side, tools are testing with a mechanism by executing the code, which is called dynamic testing. We also call static analysis as 'Source Code Analysis’. The uses of static analysis tools are such as compiler optimisation, coding support and detection of security vulnerabilities and bugs. It reports bugs such as injections, cross-site scripting (XSS), buffer overflow, and dead code. There are different techniques followed for analysing source code. One example as ’Data Flow Analysis’, it tests the source code by dividing into basic blocks. Here is an example; a php program, as seen below, is divided into blocks, and they consider each block as one node.

[frame=single]  
  
$a = 0;  
$b = 1;  
  
if ($a == $b)  
{ # start of block  
echo "a and b are the same";  
} # end of block  
else  
{ # start of block  
echo "a and b are different";  
} # end of block

After that, a path is formed by connecting the nodes along the control flow, as seen in the figure.

There are different kinds of tools available for doing static code analysis such as IDE notifications, IDE tools, dedicated tools, Linters and CLI tools. Out of which, the dedicated tools and CLI tools are more likely to report security vulnerabilities, whereas others are more likely to report coding style issues. In this thesis, we will look into the usability aspect of such tools.

## Usability

The Human-Computer Interaction researcher, Dr Jakob Nielsen, defines Usability as "a quality attribute that assesses how easy user interfaces are to use". They characterise it by five quality elements:

1. **Learnability:** It says whether the design is simple for the person seeing the design for the first time, and he can understand it.
2. **Efficiency:** After getting familiar with design, how simple is it to do the tasks in a shorter time as possible?
3. **Memorability:** Once the person knew the design and worked for some time and later when he comes back after some significant time, then is the design helpful enough to memorise?
4. **Errors:** It signifies the possibility of a person encountering the design make some errors and then whether he could recover from it quickly.
5. **Satisfaction:** It determines the delightfulness of the person using the design.

## Wireframe

A wireframe is a methodology of showing a blueprint for a particular product. For example, if we consider wireframing a User Interface of an Application. It means to show a blueprint design of how the Application UI look like with different elements such as buttons, text boxes placed on it and also how they interact and navigate to other User Interfaces. The blueprint design could vary from low fidelity, i.e., rough sketches to high fidelity, i.e., a closer look to the desired final UI. There are several tools available in the market to wireframe and Balsamiq is one such tool. It is a prototyping the actual product with a precise simulation in functionality and visualisation of the actual product to some extent. The advantage of this is the easiness of finding the mistakes in design earlier and so the cost of fixing them would be less. Here is an example of Wireframe done to a website idea about how it should look and the elements I would like to see as on an end product i.e., final front-end design of the website coded is shown in the following of what one could design using Balsamiq tool and discuss with peers how they feel or to oneself to draw the ideas of what they want to achieve before coding to get the web page with the desired UI.

An example to Wireframe a Website.

An example to Wireframe a Website.

[fig:wireframe\_website]

## User Experience Design

User Experience (UX) Design plays a vital role in the success of a product but often underestimated. In a typical software industry, they say reasons to like it might lead to over budget or no time to skip it and get right into the development phase. Well, a User Experience Design process is far beyond what we know as User Interface Design or Usability. Wireframe tool is typically used to design user interface, and usability is a way of testing whether the designed product is usable enough. A User Experience (UX) Design is a user-centred process which emphasis on the context of the user and his needs rather than focusing solely on interface design. For example, let us say someone designed a navigation application where the user says that he wants to reach from point A to point B, and it displays the route. Now, what if the user wants to mention points as a market name instead of street name and your underlying database consists only street names, that is a bad user experience (UX) even though the application is designed with better UI and also usable? Its Design cycle as seen in the illustrates an iterative cycle where we gather requirements, then we make a prototype based on the requirements and then tested, then again we gather new requirements based on the evaluation.

UX-Design.

UX-Design.

We can see the applicability of the User Experience Design in detail at .

**Likert scale**

In the process of testing the opinions of a person on a particular topic or question in specific, it is better to test how strongly they agree or not rather than stating agree or disagree. For this kind of evaluation, we use a scale called a Likert scale. Mr Rensis Likert who is American Social Psychologist, has created it and so named after him.

We use Likert scales for closed-ended questions. It helps to get a detailed analysis of the designs proposed in this thesis whether how likely they are usable in terms of user perspective instead of answering binary questions of yes or no. Likert Scales are of two types, i.e., Uni Polar and Bi-Polar.

Uni Polar Likert Scale shows one attribute let’s say strongly agree on end and I do not agree on the other end whereas for Bi-Polar Likert Scale it shows strongly agree one end and strongly disagree on the other end and also there is a label in middle which usually states ‘neither’. In this thesis work, the Uni Polar Likert Scale is used with labels 0 and 10 on either end of a scale showing ‘not at all usable’ to ‘highly usable’. Not at all usable could be understood as the worst design or solution idea for an user scenario.