**How to Guide**

**“How to: Develop your programming skills & knowledge from novice to beginner whilst learning software engineering practises involved in testing programs!”**

By Amar Mohamed

# **Acknowledgement**

Dr.Karina Rodriguez Echavarria for allowing me to attend her software verification & validation module and engage with the contents presented in the lectures and tutorials; As well as that providing me constant support and information in regards to the scope of computer science as a whole, to develop my software engineering skills.

# **Abstract**

This is a how to guide in software engineering for novices; it covers software development practises involved in testing and the techniques used. As well as that the guide has a section where the reader develops their programming skills by learning python following exercises in which they can do and follow.

In completion of the guide this will make the reader go from novice to beginner.

# **Contents**

[Acknowledgement 2](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240030)

[Abstract 2](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240031)

[Contents 2](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240032)

[List of Figures 4](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240033)

[1 Introduction 5](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240034)

[2 Regulatory bodies 6](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240035)

[2.1 ISO 29119 6](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240036)

[3 Software Development 7](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240037)

[3.1 Software Development Life Cycle Models 8](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240038)

[3.1.1 Waterfall model: 8](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240039)

[3.1.2 Agile process model: 9](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240040)

[3.1.3 Iterative and incremental development process: 9](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240041)

[3.2 Tools for enabling collaboration between teams used in Software development: 11](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240042)

[11](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240043)

[3.2.1 Version control 11](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240044)

[4 Testing Processes and Techniques 13](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240045)

[4.1 Testing Process 13](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240046)

[4.2 Testing Techniques 14](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240047)

[5 Approach to Testing 16](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240048)

[5.1 Equivalence partitioning. 17](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240049)

[5.2 Boundary Value Analysis 18](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240050)

[6 Python 19](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240051)

[Learning the Basics of Python 21](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240052)

[7 How to use Python 22](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240053)

[7.1 Variables and types: 23](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240054)

[7.1.1 Strings 25](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240055)

[7.1.2 Lists 27](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240056)

[7.2 Basic operators 29](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240057)

[7.2.1 Arithmetic Operators 29](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240058)

[7.2.2 The modulo (%) operator 30](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240059)

[7.2.3 Power relationship 31](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240060)

[7.2.4 Using Operators with Strings 32](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240061)

[7.2.5 Using lists with operators 34](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240062)

[7.3 STRING FORMATTING 35](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240063)

[7.4 Conditions 38](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240064)

[7.4.1 Boolean operators 39](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240065)

[7.4.2 “in” Operator 40](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240066)

[7.4.3 “is” Operator 41](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240067)

[7.5 Loops 42](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240068)

[7.5.1 for loops 42](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240069)

[7.5.2 While loops 44](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240070)

[7.6 break and continue statements 45](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240071)

[7.6.1“else” clause 46](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240072)

[7.7 Functions 47](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240073)

[7.7.1 return 48](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240074)

[7.7.2 Classes and Objects 49](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240075)

[7.7.3 Accessing Object Functions 51](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240076)

[8 Testing in Python 52](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240077)

[8.1 Conducting Unit Test in Python 55](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240078)

[9 Conclusion 59](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc107240079)

# **List of Figures**

[**Figure 1 -** ISO Flow Chart 6](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_Toc41831219)

[Figure 2- Waterfall Model process diagram 8](file://users/amz/Documents/Independent%20Study/How%20to%20guide%20MODDED.docx#_Toc41831220)

[Figure 3-Agile Model process diagram 9](file://users/amz/Documents/Independent%20Study/How%20to%20guide%20MODDED.docx#_Toc41831221)

[Figure 4-Iterative and Incremental Development process model 10](file://users/amz/Documents/Independent%20Study/How%20to%20guide%20MODDED.docx#_Toc41831222)

[Figure 5-This shows the structure of a test plan. 54](file://users/amz/Documents/Independent%20Study/How%20to%20guide%20MODDED.docx#_Toc41831223)

[Figure 6-Software Requirement Specification Document. 64](file://users/amz/Documents/Independent%20Study/How%20to%20guide%20MODDED.docx#_Toc41831224)

# **1 Introduction**

Over the course of this guide, you will be introduced to software engineering practises involved in software verification and validation, as well developing your programming skills from novice to beginner. This will include activities/learning on following:

Software development practises involved in testing

Python exercises

Testing in Python

# **2 Regulatory bodies**

Software testing is an important part of software development cycle, as you need to ensure that the code that had been written does what it’s required to do. Essentially the aim of the testing phase is to be able to detect defects within the code, as stated by per IEEE Standard 610.12-1990; “The process of analysing a software item to detect the differences between existing and required conditions (that is, bugs) and to evaluate the features of the software items”.

## 2.1 ISO 29119

**ISO/IEC/IEEE 29119** is a standardization which covers systems and software engineering testing. The aims of the standardization are:

To create a set of standards that are agreed internationally.

To have adaptable standards for different organizational scenarios that cope up with current technological advancement in software development.

The new international standard is envisioned to be more comprehensive by covering various level of testing processes.

**Figure 1 -** ISO Flow Chart

The software testing model is based on a fundamental testing model where processes testing forms the core of the model - ISO 29119.

The scope and structure of the ISO 29119 is a compilation of standards to give use the ISO 29119.

The scope of the standard will be covered in the guide by the activities and topics introduced to you.

# **3 Software Development**

**This chapter will introduce you to:**

Understanding of software development practises

Types of testing

Use of version control for sharing code prior to testing

**Software engineering process:**

Software development life cycle is process which is followed for a software project, within a software organization, this consists of:

**Requirements:** The first stage involves understanding what needs to be designed and what are the functions, purpose, etc.

In this stage the specifications of the input and output/ or the final product are studied and marked.

**System Design:** The requirement specs from the first stage are studied which allows you to design the system.

System Design allows you to figure out what hardware and system requirements are needed. Furthermore, System Design helps in defining the overall system architecture.

The software code to be written in the next stage is created in this stage.

**Implementation:** With inputs from the system design, the system is first developed in small programs called units.

Each unit is developed and tested for its functionality (Unit Testing)

**Integration and Testing:** All the units developed in the implementation stage are integrated into a system.

The software designed needs to go through constant testing to find out if there are any flaws or errors.

Testing is done so that the client does not face any problems during the installation of the software.

**Maintenance:** Modifications to the system or an individual component is done to improve on the systems performance.

Modifications arise either due to change requests initiated by the customer, or defects uncovered during the use of the system. The client is provided with regular maintenance and support for the developed software.

**It’s essentially 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.**

## 3.1 Software Development Life Cycle Models

There are various software development life cycle models defined and designed which are followed during the software development process.

**Models that are available for software development life cycle:**

### 3.1.1 Waterfall model:

|  |
| --- |
| Figure 2- Waterfall Model process diagram |

The **Waterfall Model** is the first Process Model to be introduced. It is very simple to understand and use. In a Waterfall model, each stage must be completed before the next stage can begin and there is no overlap between any of the phases.

Advantages of the Waterfall Model

Allows for departmentalization and control:

A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model stages one by one.

Progresses through easily as the stages are understandable and explainable and thus it is easy to use.

It is easy to manage due to the rigidity of the model – each stage has specific deliverables and a review process.

Stages are processed and completed one at a time and they do not overlap. The model works well for smaller projects where requirements are very well understood.

**Disadvantages of Waterfall Model**

*It is difficult to estimate time and cost for each stage of the development process.*

*Once an application is in the testing stage, it is very difficult to go back and change something that was not well-thought-out in the concept stage.*

*Not a good model for complex and object-oriented projects.*

*Not suitable for the projects where requirements are at a moderate to high risk of changing.*

### 3.1.2 Agile process model:

The meaning of Agile is swift.

**Agile process model** is based on iterative development.

Agile methods break tasks into smaller iterations, or parts that do not require long term planning.

The project scope and requirements are stated at the start of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly stated at the start.

|  |
| --- |
| Figure 3-Agile Model process diagram |

The entire project is grouped into smaller parts which helps to minimize the project risk and to reduce the overall project delivery time requirements; as well as that each iteration involves a team working through a full software development life cycle. Each iteration is considered as a short time "frame" in the Agile process model, which lasts from one to four weeks.

### 3.1.3 Iterative and incremental development process:

Iterative and incremental development process is one of the methodologies of Agile software development, rational unified process and extreme programming. Based on producing deliverables, different parts of the system are developed at various times or rates and are integrated based on their completion.

Teams plan to revisit parts of the system in order to revise and improve them whilst consulting users for feedback.  
   
This process model came about due to the flaws in the waterfall model.

|  |
| --- |
| Figure 4-Iterative and Incremental Development process model |

It differs from the waterfall model because it is cyclical rather than unidirectional, which allows greater ability to add changes into the application during the development cycle.

## 3.2 Tools for enabling collaboration between teams used in Software development:

Project management tools

Communication tools

Version control tools

### 

### 

### 

### 3.2.1 Version control

Version control is the practice of tracking and managing changes to code. Version control managements system provides a running history of code development and allows many people to work in parallel. This helps to streamline the development process and provide a centralized source for all codes.

**Git - version control tool:**

Git is a popular version control tool.

**Using Version control tool:**

**Repository:**

Is a folder for your project.

Contains all of the projects files and stores the files revision history

Can own repositories individually or share with other people involved in the project.

How to use version control tools varies on the tool being used. Look at the websites for your chosen version control tool that you will use when undertaking a software developing project.

# 

# **4 Testing Processes and Techniques**

**This chapter will introduce you to:**

Overview of testing processes and techniques

Types of testing techniques

## 4.1 Testing Process

The testing process involves 5 main steps but before you follow them you need to gather information to understand the requirements of the program and develop a document highlighting them **(first step).**

Once that’s complete you will need to:

Derive all important **scenarios.**

Design **test scenarios** for every scenario that can happen running the program.

Assign all planned test scenarios to different **test cases.**

|  |
| --- |
| **An example of a test case:** |

Choose a **test design technique** for each requirement.

## 4.2 Testing Techniques

**Types of Testing Techniques:**

**Unit Testing:** Individual units or modules are tested by the developer

**Integration Testing:** Individual modules are grouped together and tested by the developers.   
The purpose of this is that it allows you to determine what modules are working as expected once they are integrated.

**System Testing:** Performed on the whole system by checking whether the system or application meets the requirement specification document.

2 main categories of testing techniques:

**Static** – Code isn’t executed

**Dynamic** – Code is executed

|  |
| --- |
| **Dynamic testing is split into 2 types of testing.** |

|  |
| --- |
| **WHITE BOX TESTING:**  AKA – Structural testing/Glass box testing  Takes into account the internal mechanism of a system or component  Mainly performed by the developers or white box testers that have programming knowledge. |

|  |
| --- |
| **BLACK BOX TESTING:**  AKA – Functional testing  This testing method ignores the internal mechanism of the system or component  Verifies the functionality of the system under test  Focus’ on the outputs generated in response to selected inputs and execution conditions  Mainly performed by the tester and requires no prior programming knowledge |

# **5 Approach to Testing**

**This Chapter will introduce you to:**

Reflecting on what needs to be tested and how

The chapter will cover how a software engineer would go about testing his software to give you an insight to the approach of testing.

**First Cycle of Testing:**

**Example:** You have a simple program that receives two 1- or 2-digits numbers from the input and multiplies them together

You start running the program and test whether it exactly multiples two given numbers or not. **Then ask yourself**:

Does the program work?

Are there any errors?

Does this mean the program/application is ready to be released?

**Steps for Testing:**

**Familiarise yourself with the program:** Understanding what it is and what it’s supposed to do.

**Check what needs to be tested:** Checking what needs testing.

**Perform Tests**

**Summary:** Summarise what you know about the program and its problem.

**FAMILIARISING YOURSELF WITH THE PROGRAM:**

**Familiarise:** Inspect the code

**Check:** Check if the program is stable enough to be tested

Not if the programs sent for formal testing is crashing without reasons.

**Initially you might be able to notice problems/errors**

Design errors (This depends on the type of program)

1) Are there any onscreen instructions?

2) Any information on screen showing you are on the right program?

3) Can you stop the program?

**Coding errors:**

Does everything look coded properly?

**2)CHECK WHAT NEEDS TESTING:**

**Think of valid and invalid input values you will need to input to test the program.**

**Using the example program- Inputs to test the program:**

1- or 2-digit numbers

What are the biggest numbers that can be multiplied?

What will happen if a user enters 3-digit numbers?

What about negative numbers?

What about zero?

Programs often fail on zero.

**By thinking of valid and invalid input values to input to test the program this produces a series of formal tests that you can use later.**

**This will be a well-documented group of tests that you will use each time testing a new version of the program.**

**SELECTING TEST CASES:**

**Choosing tests to run:** There are too many possible tests to run therefore pick those test cases that are significant.

**Test results:** If you expect the same results from two test cases, they belong the same category/class of test cases as you expect the same result from the two test cases.

Once you have realised that you are dealing with a class of test cases this allows you to test a few and ignore the rest.

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 5.1 Equivalence partitioning

In Equivalence Partitioning (Black box testing technique [4.2](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_4.2_Testing_Techniques)) input values to the system are divided into different groups based on the similarity of the outcome. This reduces the total number of test cases to a finite number of testable test cases, whilst ensuring requirements are met.[9][10]

**In testing you would make 2 equivalence partitions** – valid and invalid

For example:

Let's consider the behaviour of an Order Burger Text Box

Burger values 1 to 10 is considered valid. A success message is shown.

While value 11 to 99 are considered invalid for order and an error message will appear, "Only 10 Burgers can be ordered"

**Here is the test condition**

Any Number greater than 10 entered in the Order Burger field (let say 11) is considered invalid.

Any Number less than 1 that is 0 or below, then it is considered invalid.

Numbers 1 to 10 are considered valid

Any 3 Digit Number say -100 is invalid.

We cannot test all the possible values because the number of test cases will be more than 100. You use equivalence partitioning where you divide the possible values of tickets into groups or sets as shown where the system behaviour can be considered the same.

**Valid partition:**

inputs 1 – 10

**Invalid partition:**

Lesser than the range inputs, greater than inputs, alphabets data inputs.

Essentially divides the test conditions into groups based on an equivalence criterion

## Boundary Value Analysis

The best test cases are at the boundaries of a class.

Boundary Value Analysis Testing technique (Black box testing technique [4.2](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_4.2_Testing_Techniques)) is used to identify errors at boundaries, it’s the next part of Equivalence Partitioning for designing test cases where test cases are selected at the edges of the equivalence partitions ([5.1](https://word-edit.officeapps.live.com/we/wordeditorframe.aspx?new=1&ui=en-US&rs=en-GB&hid=oDOEnzQZhU2BsE7FGm3EXQ.0&wopisrc=https%3A%2F%2Fwopi.onedrive.com%2Fwopi%2Ffiles%2F3044E950A3756AF0!541&wdnewandopenct=1656428372309&wdprevioussession=6b8a83f6-0075-458d-ada9-7b9def138c84&wdorigin=Unknown&wdo=2&wde=docx&sc=host%3D%26qt%3DDefault&mscc=1&wdp=0&uih=OneDrive&jsapi=1&jsapiver=v2&corrid=f87c183c-2f88-4ded-b93c-54a28489c8e4&usid=f87c183c-2f88-4ded-b93c-54a28489c8e4&newsession=1&sftc=1&wdredirectionreason=Unified_SingleFlush#_5.1_Equivalence_partitioning)).

So, these extreme ends like Start- End, Lower- Upper, Maximum-Minimum, Just Inside-Just Outside values are called boundary

The basic idea in boundary value testing is to select input variable values at their:

Minimum

Just above the minimum

A nominal value

Just below the maximum

Maximum

# **6 Python**

**This chapter will introduce you to:**

Python as a developing framework

**Python:**

Python is an interpreted general-purpose language.

**Python Integrated Development Environment(IDE):**

Python has many IDE’s as they are free/open source scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts

**Spyder User Interface:**

One of the main IDE used to code in python

**Python Interpreter:**

Python isn’t compiled but is interpreted at runtime.

The Python interpreter is usually available from the console

**Source code encoding:**

Source files are encoded in UTF-8, therefore to use an encoding that isn’t the default one, a special comment line is added as the first line of the file.

**The syntax as follows:**

# -\*- coding: encoding -\*-

**By default you will see the UTF-8 encoding on the IDE:**

# -\*- coding: utf-8 -\*-

**Executable code for console:**

Python scripts can be made directly executable by putting the line - #!/usr/bin/env python3.5

**Learning python:**

Learnpython.org

Wiki.python.org/moin/BeginnersGuide

# **Learning the Basics of Python**

In this section of the guide we will be looking at tutorials for you to follow and complete to develop your programming skills from novice to beginner.

These exercises have been adapted from [https://www.learnpython.org](https://www.learnpython.org/)

Python is a very simple language with a very straightforward syntax.

You can access Spyder user interface by downloading anaconda navigator <https://www.anaconda.com/products/individual>

Python IDE – Spyder:

# **7 How to use Python**

The simplest directive in Python is the "print" directive - it simply prints out a line.

This section explains how to use the basic “print” command in Python.

Using “print” command to print: hello, world!

## 7.1 Variables and types:

This section explains how to use basic variables and types in Python.

The basic variables In python are strings and number

You do not need to declare variables before using them, or declare their type. Every variable in Python is an object.

Python supports two types of numbers - integers and floating point numbers.

To define an integer, use the following syntax- **exercise**:

myint = 7

print(myint)

To define a floating point number, you may use one of the following notations- **exercise**:

myfloat = float(9) or myfloat = 7.5

print(myfloat) print(myfloat)

## 

### 7.1.1 Strings

Sentences are stored in Python as strings. This means that string means sentence.

Strings are defined either with a single quote or a double quotes. For example:

To print a string in Python 3:

mystring = ‘hello, world!’

print(mystring)

or

mystring = “hello, world!”

print(mystring)

There are additional variations on defining strings that make it easier to include things such as carriage returns, backslashes and Unicode characters. <https://docs.python.org/3/tutorial/introduction.html#strings>

Assignments (assigning values) can be done on more than one variable "simultaneously" on the same line.

**For example:**

a, b = 3, 4

print(a, b)

Exercise: different definitions of strings being executed to show different ways you can get the same sentence.

### 7.1.2 Lists

This section explains how to use lists in Python.

Lists are very similar to arrays. They can contain any type of variable, and they can contain as many variables as you wish.

Syntax for lists **exercise**:

Mylist = []

for x in mylist:

print(x) - This syntax lets you print all the variables in list format.

Note: The index is zero based. Therefore if you want to access the second item on the list its index is 1.

**Exercise:** Add numbers and strings to the correct lists using the "append" list method.

You must add the numbers 1,2, and 3 to the "numbers" list, and the words 'Amar' and 'cool' to the strings variable

## 7.2 Basic operators

This section explains how to use basic operators in Python.

### 7.2.1 Arithmetic Operators

The addition, subtraction, multiplication, and division operators can be used with numbers.

Syntax for using arithmetic operators’ **exercise:**

number = 1+1+2\*3/10

print(number)

### 7.2.2 The modulo (%) operator

The modulo (%) operator, which returns the integer remainder of the division.

Syntax for using the modulo operator **exercise**:

remainder = 15 % 1.2

print(remainder)

## 7.2.3 Power relationship

Using two multiplication symbols makes a power relationship.

Syntax for power relationship **exercise**:

xx = 7\*\*2

xx = 15\*\*3

print(number1)

print(number2)

### 7.2.4 Using Operators with Strings

Python supports concatenating strings using the addition operator.

Syntax for concatenating strings using the addition operator **exercise**:

HiGoodbye = “Hi”+” ”+”Goodbye”

**Python also supports multiplying strings to form a string with a repeating sequence:**

Syntax for forming a string with a repeating sentence **exercise:**

lotsofeverything = “lots of everything”\*10

print(lotsofeverything)

Just like previously, forming new lists with a repeating sequence using the multiplication operator:

**Exercise:**

print([10,20,30]\*3)

7.2.5 Using lists with operators:

Lists can be joined with the addition operator.

Syntax for adding lists together **exercise**:

even\_numbers = [2,4,6,8]

odd\_numbers= [1,3,5,7]

all\_numbers = even\_numbers + odd\_numbers

print(all\_numbers)

## 7.3 STRING FORMATTING

The "%" operator is used to format a set of variables enclosed in a "tuple" (a fixed size list).

Using a format string, which contains normal text together (simple text) with "argument specifiers", special symbols - "%s" and "%d".

**Exercise**:

Variable called "name" with your user name in it, and you would then like to print( a greeting to that user.)

**Exercise:**

name = “Scott”

print(“Hello, %s!” % name)

**two or more argument specifiers:**

The **exercises** shown below shows the use of two or more argument specifiers. When there are two or more argument specifiers use a tuple.

**Exercise 2:**

## 7.4 Conditions

Python uses Boolean variables to evaluate conditions. The Boolean values True and False are returned when an expression is compared or evaluated.

variable assignment is done using a single equals operator "=".

Comparison between two variables using double equals operator ‘==’

The “not equals” operator is marked as “!=”

**An exercise of the use of a Boolean variable is shown below.**

**Exercise:**

x = 2 - integer 2 is assigned to the variable x

print(x == 2) – this prints out True as an expression of 2 is evaluated

print(x == 3) – this prints out false as the expression of 3 is evaluated and isn’t equal to the variable x’s assignment (which is 2)

print(x < 3) - This prints out true as x is less than 3

### 7.4.1 Boolean operators

### 7.4.2 “in” Operator

“in” operator :is used to check if a specified object exists within an iterable object container (i.e. a list).

**Exercise:**

### 

### 

### 

### 7.4.3 “is” Operator

the "is" operator does not match the values of the variables, it is the variables themselves.

**Exercise:**

Exercise 2:

## 7.5 Loops

There are 2 types of loops.

for and while.

### 7.5.1 for loops

for loops iterate for a given sequence.

Syntax to initiate for loop iteration **exercise**:

primes = [1,3,5,7,9]

for prime in primes:

print(prime)

for loops can iterate over a sequence of numbers using the “range” function.

for loop to iterate over a specific sequence of numbers requires the range function **exercise**:

for x in range(9):

print(x)

**Exercise 2:**

### 7.5.2 While loops

While loops repeat as long as a Boolean condition is met

This example shows that the condition for how this works:

count = 0 – assigning 0 as variable count

while count < 5: - this means that the loop is repeated for integers between 0 and less than 5

print(count) – syntax to execute the command.

count = count +1- this is necessary as this makes sure the program understands that from 0 you add on 1. Once reaches 4 conditions has been met

**Exercise 2:**

## 7.6 break and continue statements

Exercises below cover break and continue statements:

**Exercise 1:**

**Exercise 2:**

### 7.6.1“else” clause

When the loop condition of “for” or “while” statement fails then the code part of else is executed

Exercises below cover how “else” clause are executed when statements fail:

**Exercise 1:**

**Exercise 2:**

if **break** statement is executed inside for loop then the "else" part is skipped.

## 7.7 Functions

Functions are a way to divide codes into useful blocks. This allows an order for codes making it easier to read and reuse.

Functions are a keyway to define interfaces so code can be shared easily.

Functions may also receive arguments (variables passed from the caller to the function)

function's name followed by (), placing any required arguments within the brackets

calling the function’s **exercise**:

def my\_function(): - defining the function using def

print(“Hello From My Function!”) - defining what we want our function to print

my\_function() – this our function that we defined to print out the greeting

def my\_function\_with\_args(username, greeting): - defining the function with arguments

Print("Hello, %s , From My Function!, I wish you %s"%(username, greeting)) - defining what we want our function to print

my\_function\_with\_args("Amar Mohamed", "a great year!") – prints out Hello, Amar Mohamed, From My Function!, I wish you a great year!"

### 7.7.1 return

**return** is used to return a value to the caller.

(To "**call**" **means:** To make a reference in your code to a function that is written elsewhere.)

returning value/string to the caller is essentially returning your reference in your code to a function that’s written somewhere else, how that’s done is shown in the **exercise** below:

**Exercise:**

### 7.7.2 Classes and Objects

Objects are an encapsulation of variables and functions into a single entity. Objects get their variables and functions from classes. Classes are essentially a template to create your objects.

to assign the class(template) to an object and access it you:

**Exercise 1:**

class MyClass:

variable = "blah"

def function(self):

print("This is a message inside the class.")

myobjectx = MyClass()

print(myobjectx.variable

**Creating multiple different objects that are of the same class:**

**Exercise 2:**

class MyClass:

variable = "blah"

def function(self):

print("This is a message inside the class.")

myobjectx = MyClass()

myobjecty = MyClass()

myobjecty.variable = "yackity"

print(myobjectx.variable)

print(myobjecty.variable)

### 7.7.3 Accessing Object Functions

To access a function inside of an object you use notation similar to accessing a variable.

**Exercise:**

class MyClass:

variable = "blah"

def function(self):

print("This is a message inside the class.")

myobjectx = MyClass()

myobjectx.function()

# **8 Testing in Python**

**UNIT TESTING:**

Python unittest as a unit/integration testing tool is used to implement both black and white box tests.

**PYTHON UNIT TESTING FRAMEWORK:**

Known as PyUnit. The module unittest in python supports test automation and aggregation of tests into collections.

**STRUCTURE OF TEST PLAN:**

|  |
| --- |
| Figure 5-This shows the structure of a test plan. |

**TestSuite:** A collection of test cases, test suites, or both.Used to aggregate tests that should be executed together.

**TestCase:** Smallest unit of testing – checks for a specific response to particular set of inputs.

Unittest provides a base class, TestCase which is used to create new test cases.

**AN EXAMPLE OF UNIT TESTING IN PYTHON:**

This is an example of a unit test in python that tests a ‘for loop’ program that receives two 1- or 2-digit numbers from the input and multiplies them together without using the star operator.

## 8.1 Conducting Unit Test in Python

There are 3 functionality tests being conducted in this program

The 3 tests each check for a specific response to a particular set of inputs.

testTwoPositiveNumbers – Test checking that multiplication of 8 and 7 give 56

test1PositiveNumber-Test checking that multiplication of 8 and 5 give 40

testNegativeNumbers - Test checking that multiplication of - 8 and 5 give – 40

This method runs all the tests together at the same time indiscriminately.

**In this exercise you are creating a test suite to allow individual tests to be aggregated – this only allows tests added into the suite to be tested:**

**In this exercise you are creating another test suite to show how test suites can be aggregated, which will allow you to test, test cases in another test suite:**

**In this final exercise you are running tests for 2 test suites in which you have selected test cases to test:**

# **9 Conclusion**

Congrats! You have now completed “How to: Develop your programming skills & knowledge from novice to beginner whilst learning software engineering practises involved in testing programs!” with this now you should have built up enough courage to dive into the world of computer science as a beginner.

Now you can start looking at completing the **ADVANCED** tutorials in Python to further develop your programming skills which can be found at [https://www.learnpython.org](https://www.learnpython.org/) ; with that you can develop a program that you can manage and test to further apply the practises you have adopted.

You can also look into the different other kinds of topics that weren’t covered in this guide such as the many types of testing that are around (<https://www.softwaretestinghelp.com/types-of-software-testing> ) or Bug reports – how to write them and their importance (<https://www.softwaretestinghelp.com/how-to-write-good-bug-report>).

# **References**

[1] Admin, S., 2020. *Boundary Value Analysis and Equivalence Class Partitioning With Simple Example*. [online] Software Testing Class. Available at: <https://www.softwaretestingclass.com/boundary-value-analysis-and-equivalence-class-partitioning-with-simple-example/> [Accessed 27 April 2020].

[2] Bronathan, B. and verma, s., 2020. *What Is Boundary Value Analysis And Equivalence Partitioning.* [online] Softwaretestinghelp.com. Available at: <https://www.softwaretestinghelp.com/what-is-boundary-value-analysis-and-equivalence-partitioning/> [Accessed 27 April 2020].

[3] Dr. Karina Rodriguez Echavarria, “Verification and validation, lecture 2: testing in the context of software development” 10.02.2020. [Online.]. Available: <https://learn-eu-central-1-prod-fleet01-xythos.s3-eu-central-1.amazonaws.com/5d108c67a3df7/3840632?response-content-disposition=inline%3B%20filename%2A%3DUTF-8%27%27CI316.02.TestinginSoftdevelopment.pdf&response-content-type=application%2Fpdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20200323T100623Z&X-Amz-SignedHeaders=host&X-Amz-Expires=21600&X-Amz-Credential=AKIAZH6WM4PLYI3L4QWN%2F20200323%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Signature=727b9e6fa5fba645e66cbdc473aa10d9f0ad6731619c4524c81fe5d571049bc8> [Accessed 10/02/2020].

[4] Dr. Karina Rodriguez Echavarria, “Verification and validation, lecture 3: Source control for testing” 17.02.2020. [Online.]. Available:

<https://learn-eu-central-1-prod-fleet01-xythos.s3-eu-central-1.amazonaws.com/5d108c67a3df7/3874924?response-content-disposition=inline%3B%20filename%2A%3DUTF-8%27%27CI316.03.SourceControl.pdf&response-content-type=application%2Fpdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20200327T150350Z&X-Amz-SignedHeaders=host&X-Amz-Expires=21600&X-Amz-Credential=AKIAZH6WM4PLYI3L4QWN%2F20200327%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Signature=8af3c3e96fccc94ccb4a2d12bf53865c080a96a4d0c1531329debca6d8273ab3>

[Accessed 10/02/2020].

[5] Dr. Karina Rodriguez Echavarria, “Verification and Validation, Lecture 4: Testing techniques 24.02.2020. [Online.] Available:

<https://learn-eu-central-1-prod-fleet01-xythos.s3-eu-central-1.amazonaws.com/5d108c67a3df7/3911726?response-content-disposition=inline%3B%20filename%2A%3DUTF-8%27%27CI316.04.PythonTestingDevelopment.pdf&response-content-type=application%2Fpdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20200407T092415Z&X-Amz-SignedHeaders=host&X-Amz-Expires=21600&X-Amz-Credential=AKIAZH6WM4PLYI3L4QWN%2F20200407%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Signature=73a0f0419aee3ea5ddeb3311bbc434896cb95fa9afeeb451201f5306f6178967>

[Accessed 27/02/2020].

[6] Dr. Karina Rodriguez Echavarria, “Verification and Validation, Lecture 5: Python for development and testing” 09.03.2020. [Online.] Available:

<https://learn-eu-central-1-prod-fleet01-xythos.s3-eu-central-1.amazonaws.com/5d108c67a3df7/3948429?response-content-disposition=inline%3B%20filename%2A%3DUTF-8%27%27CI316.05.PythonTutorial.pdf&response-content-type=application%2Fpdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20200408T080631Z&X-Amz-SignedHeaders=host&X-Amz-Expires=21600&X-Amz-Credential=AKIAZH6WM4PLYI3L4QWN%2F20200408%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Signature=167de28293fdb8709309b9e8df00604081e565d8fdc8d550e29bb03d7b6442f6>

[Accessed 10/03/2020].

[7] Dr. Karina Rodriguez Echavarria, “Verification and Validation, Lecture 6: Cycles of Testing” 11.03.2020. [Online.] Available: <https://learn-eu-central-1-prod-fleet01-xythos.s3.eu-central-1.amazonaws.com/5d108c67a3df7/4021775?response-cache-control=private%2C%20max-age%3D21600&response-content-disposition=inline%3B%20filename%2A%3DUTF-8%27%27CI316.07.PythonUnitTest.pdf&response-content-type=application%2Fpdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20200601T120000Z&X-Amz-SignedHeaders=host&X-Amz-Expires=21600&X-Amz-Credential=AKIAZH6WM4PLYI3L4QWN%2F20200601%2Feu-central-1%2Fs3%2Faws4_request&X-Amz-Signature=a68359cf2b3f330cda6d93a2d4e4a38e1184a6f7966f773b0413476997170967>

[Accessed 15/03/2020].

[8] Dr. Karina Rodriguez Echavarria, “Verification and Validation, Lecture 7: Unit Testing”

18.03.2020. [Online.] Available: