

Software Testing: Presentation notes

Scope

Scope in reference to the background and problem setting

Referenced section	Defined Scope	Reasoning
"4-16 weeks for testing."	Test what is possible in the time frame	The decided time frame for testing (within the 4-16-week scope) will be decided based on the amount (and amount of detail) of testing on the agenda. The testing length will be decided on how long is estimated to be needed to implement all desired testing. If that estimated time would exceed the 16-week cap, then the amount of testing will have to be cut down to the point where the amount remaining can be done within the allowed testing time frame.

Scope in reference to requirements

Requirements No.	Defined Scope	Reasoning
1	Develop these functionality	are the functions desired by the product owner, thus are of paramount importance. Steps should be taken to implement these as best as possible. More effort into perfecting these functions rather than implementing other functions not specified. Saved time and cost due to limited functionality needs (subsequent development). The saved time and limited functions will allow for more quick testing of functions, speeding up time to market.
2	Integrate only the following systems	are the only external systems that are specified as required for existing system to function as intended. time and cost, only needed to integrate a finite amount of systems. All of which are noteworthy and commonly used systems, increasing the likelihood of familiarity amount developers, saving time and cost in integration; making the integration testing phase smoother.
3	Only account for employee and contractor data from the past 10 years	data is implicitly deemed irrelevant by the specification of 10 years. Out of the scope of their care. time of migrating over all the company's data history and having to consider data past 10 years.
4	Security and permission levels	or data access permissions can be based on the initialised job role of the user. No need to add other parameters to determine access rights

	defined by job role	time and cost needed to decipher logic needed to restrict permissions since it is only based on one parameter. When determining/testing if a user should have access, only need to consider said parameter
5	Use manual key entering data in phase 1	<p>ed usage of the swivel chair interface. No need to develop a software interface for data input, for phase 1 of the project.</p> <p>cost in needing to develop data input software but hinders time of inputting data; as its all manual entry. However, there's no need to worry about system errors messing up data when inputted since its all manual; only need to account for human error (typos).</p>
6	Application behaviour must differ for each employee type	<p>our of the system will differentiate based on what kind of user is using the system; thus based on the initialized user role, the system needs to respond accordingly. But only needs to account for the finite employee types of Nightingale Services, no more.</p> <p>the time of development, needing to create variations in the system for each user type. However, the finite amount of employee types does provide a limit to the time and cost needed to develop. Must test if the correct behaviour fits the correct user type, for all desired user types</p>
7	Only have servers active 17 hours of the day	<p>cation states that the application only needs to be active from 7am to 12am, meaning servers don't need to run the application between 12:01am and 6:59am.</p> <p>cost on server management. No need to monitor and test activity on the application in its inactive hours, saves time.</p>
8	Develop mobile version of software for apple users only	<p>d operating system is iOS, the operating system exclusively for apple products</p> <p>development cost and subsequent time to develop not having to worry about other operating systems. Also can speed up time to market. No need to consider the logic of other operating systems and using different frameworks to test them, saves time and cost in developing those OS specific tests.</p>
9	Analyse and develop the system based on the Australian legislative laws	is an Australian application, it needs to oblige by the country's laws. Being void of use outside of Australia eliminates the need to think about the laws of other nations that could affect the desired functionality of the system.

		ime needing to consider other legislation and ensuring the application doesn't conflict with any other nation specific laws.
--	--	--

- Assumptions
 - Requirement 5 – the swivel chair interface is being used exclusively for phase 1 of the project, afterwards; software will be developed to process data input
 - Requirement 8 – applying for leave on **iOS** devices insinuates that all mobile functionality of the system is iOS exclusive