Aston University
BIRMINGHAM UK

College of Engineering & Physical Sciences
Assignment Brief

DC3270 – Testing and Reliable Software Engineering	“Happy Health” System

Luis Hernan Garcia Paucar   garcial2@aston.ac.uk
Diana Galiakhmetova   190199616@aston.ac.uk

“Happy Health” System

1. Descriptive details of the assignment:

Write a report (2,500 words, +/- 10%, not including appendices) detailing a test plan, with example test cases, based on the requirements (a.k.a. features) of the “Happy Health” System **(HHS)** described in the scenario in section 2. To write the test plan, you are expected to choose a feature or a set of features with significant complexity, requiring a significant set of test cases.

The coursework must be completed **individually**, which means your submission must evidence your own thinking and understanding.

In your submission, you should provide:

1) (500 words) An overview of the test strategy, which addresses the following points:
   a) Why testing of this particular system is important.
   b) The point or points during development at which different testing approaches would take place, and the roles of anyone involved.
   c) How the system, specifically in terms of architecture, might be developed in a way that facilitates easy application of good-quality testing.
2) (2,000 words) A discussion of test strategies and techniques that you would apply in order to maximise the quality of at least one core feature of the **HHS**. You are expected to provide actual test cases, including test data and expected outcomes in each of the following sections:
   a) A discussion of black box testing strategies, supported by BDD scenarios and associated methods in a programming language of your choice.
   b) A discussion of white box testing strategies, supported by pseudocode snippets, to illustrate how such strategies would be applicable to the **HHS**.
   c) A discussion of two applicable quadrant 3 tests, including any resources that would be required to conduct such tests.

d) A discussion of two applicable quadrant 4 tests, including any resources that would be required to conduct such tests.

No code is provided for the "Happy Health" System **(HHS),** and you are not required to write the code. You should be able to develop a test plan and test cases without access to the source code. You may make assumptions about a hypothetical implementation, and you may write pseudocode where necessary to develop tests.

2. Scenario

HHS is a new system, that you will be testing and producing a test plan for. It is designed to support a variety of benefits across a wide range of users. There are 3 main internal users (and 1 external user) of this system; Doctors who are able to access patient records, add new tests, upload test results, add new equipment to inventory and order ; Nurses who can book tests on authorisation, view test results , book consultation ; Receptionist overseeing the process of adding and removing new patients and the external user is the end user of this system who can create and access their profile, request prescriptions, request access to their medical records and book appointments.

The requirements of the system are as follows:

1) The ability for a patient to register as new user on the **HHS** using the following details:
   a) Name (must be at least 2 characters)
   b) Mobile Phone (must be exactly 11 digits)
   c) Age (must be equal to or above 18 to create a new profile)
   d) Postcode (must be at least 5 characters)
   e) Email (must be checked for appropriate domain name)
   f) Previously registered as patient elsewhere (There are 2 possible choices; 0 – yes, previously registered, 1- no, this is the first time)

2) The **HHS** has a "submit application" function that will take only "ONE" application per user every 24 hours and in that process will store all the records in a database. The "submit application" function must not overwrite any existing applications (that is if the user decides to submit another application within the 24-hour period, it will not be accepted).
3) The ability for receptionist to verify and create a new patient profile using details from (1) and assign a unique medical ID, which can be anywhere between 8 and 16 digits (only alphanumeric characters). Once verified, a patient will be sent an email confirmation and text message.

4) The ability for registered doctors to see new patient information and book first appointment. Confirmation will be sent as email and text message.
5) The ability for the system to allow patients to pay for consultation, tests, and other services as deemed appropriate.
6) Registered patients must also be able to:
   a) Use the in-built journal, which takes entries in a date/time/text format, where patients can enter any notes, follow-up questions or reminders. Patient may choose to allow access to registered doctors and nurses.
   b) Book an appointment to visit the doctor or the nurse, with their unique medical ID, using a queueing system, which books appointments on a first come first-serve basis.
   c) Book urgent appointments or request for emergency care through quoting their unique medical ID by directly contacting reception.
   d) View booked appointments with doctors and/or nurses, see test results and prescriptions that have been made available to them.
   e) Request prescriptions, through doctor's authorisation. Repeat prescriptions are available, under the discretion of the doctor.
   f) Leave anonymous reviews within the system, which can take 1 entry per patient every 48 hours. Reviews can either be a number-based rating system (1 meaning poor service, 5 meaning happy health service) or a text-based review (takes a combination of alphanumeric and special characters).
7) Nurses and doctors must have the ability to:
   a) Provide prescriptions for patients (at least one appointment should be completed by the patient) via the app that can only be accessed by the doctor, nurse or patient requesting it.
   b) Authorise medical tests for patients (at least after one appointment has been completed) and upload test results securely via the system that only nurses, and respective patients have access to.
   c) Update inventory list by systematically going through the previously placed orders and verify usage levels periodically.
   d) Request for service of medical equipment based on needs. There are 3 main equipment that needs servicing on a regular basis and Happy Health would like to automate the request for servicing periodically:

Name of equipment	Authorisation level	Service due
Defibrillator	Doctor and Nurse	Every 2 years
General X-ray unit	Doctor only	Every year

X-ray viewer (single and double screen)	Doctor and nurse	Every 3 years

Table 1 – List of medical equipment and their servicing needs

8) Doctors and nurses do not have permission to delete or modify patient records (personal details).
9) The system needs to be continually updated and monitored every 3 months to ensure there is no data leakage.

> ***DISCLAIMER*** – This piece of software is by no means representative of a real-world application and has been purely designed for the purposes of this assessment, while this may have flaws and gaps, you are not expected to know how the full-fledged system will work nor build such a system. The emphasis is solely on planning appropriate tests. Only work on the requirements given and asked of you.

Submission Structure:

- The report should follow standard academic report principles, here is a recommended structure, you do not need to strictly adhere to it, perhaps can be used as a suitable starting point
  - Testing overview, describing the nature and relevance of testing (Section 1: items: 1) a), 1) b) and 1) c).
  - Test plan, subdivided into:
    - Black box testing and white box testing
    - Quadrant 3 testing
    - Quadrant 4 testing

  Note: Test plan content details in Section 1: items 2) a), 2) b), 2) c) and 2) d).

  - References *(not counted within the 2500-word limit)*
  - Appendices: Include all test cases relevant to your testing techniques here and other details pertinent to your report, you should supply examples of the test cases in the form of screenshots/figures within the main body of the report and link to the appendix section to view remaining test cases *(not counted within the 2500-word limit).*

Recommended reading/ online sources:

- Core and further reading section for individual units of the module available on Blackboard

- Support Videos for quick recap of each unit's content available under respective units on Blackboard
- Tutorial materials for respective units presented on Blackboard (including tutorial notes)

Key Dates:

These are the key dates surrounding the assignment.

21/03/2022	Coursework set
16/06/2022	Coursework Q&A drop-in session 1
28/07/2022	Coursework Q&A drop-in session 2
18/08/2022	**Submission date and time (Online only)**
08/09/2022	Expected feedback return date

Submission Details:

- A Turnitin link will be provided in the Assessment Submission section on Blackboard
- All reports must be submitted online only, no hard copies are needed.
- Please state the word count (that is excluding appendices and references) in the front page of your report. Failure to adhere will result in a 10% penalty of the mark. **Only PDF documents must be submitted.**
- Document Title must preferably be named *Firstname_Lastname_DC3270_Report*
- Only a single document must be submitted, in an easily readable font (11pt minimum). Any references and citation must clearly follow the Harvard or APA style.

A quick checklist for you before you submit:

- Have you used the correct formatting?
- Are you within the recommended word limit?
- Have you cited and referenced correctly?
- Have you included all 4 agile testing quadrants and reliability measurements in your testing plan?
- Are your appendices, tables, figures labelled correctly and included in-text apropos?

Marking Rubric:

A marking rubric for this assignment is provided below:

The nature and relevance of testing (15%)

50	The relevance of testing has been validly applied to the context of the case study; the roles of individuals within the testing process have been accurately described, and the place of testing within a software development lifecycle has been identified. Consequences of failing to test the case study system are described; roles of individuals involved in testing are accurately described, and the place of testing within the development process is clearly described.
51-59	Some reference is correctly made to specific testing approaches and why they are applicable to the scenario, together with consequences of failing to test. A range of stakeholders are identified that would contribute to the testing process. The place of testing within the development process is clearly described.
60-69	Multiple quadrants of the agile testing methodology are incorporated into the explanation of the need for testing in this context, together with consequences of failing to test. A range of relevant participants in the testing process are identified as specific to this context. The place of testing within the development process is described with accuracy and depth.
70-79	The role of a range of different testing strategies is discussed in the context of the case study, including potential consequences of failing to test. Sound acknowledgement is made of the variability that can exist between different testing teams, and the place of testing within multiple development methodologies is described.
80 +	The role of testing in this context is discussed with insight and clarity, including potential consequences of failure to implement a range of relevant specific test types. The makeup of test teams is discussed, with potential differences between them explained. The place of testing within multiple software development methodologies is assessed.

Testing and software architecture (15%)

50	Valid facets of software architecture are described in the context of how they might aid testing of a system. At least one example of sound architectural design is drawn from a reasonable implementation of the scenario system, and forms the basis for an explanation of how testing might be made easier.
51-59	Multiple examples of sound architectural design are drawn from a reasonable implementation of the scenario system, or a single example is developed to the point that it could be implemented by a competent third party.
60-69	Examples drawn from the scenario system are relevant and would make a substantive impact upon the ease with which quality testing could be implemented. Explanations are of such a quality that suggestions could be implemented by a competent third party

70-79	Key aspects the scenario system are used as a means of explaining how sound architectural design can facilitate testing. Examples used would be in line with expectations of a professional tester.
80 +	The role of system architecture in testing is discussed insightfully, with relevant context-specific examples that would be in line with or an improvement on current best practice.

Black box testing and BDD (25%)

50	A range of valid test cases are provided that offer a broad understanding of the testing strategy, and valid insight is offered into the selection of the test cases as well as the strategy as a whole. Errors within Cucumber steps and the corresponding implementation language are permitted at this level, provided that the code could still be implemented by a competent third party.
51-59	An appropriate testing strategy has been followed, with test cases that are, on the whole, executable and in keeping with the test strategy. Justification of the test strategy and test cases are consistently relevant to the scenario system.
60-69	Test cases adhere to a well-selected test design strategy. Accurate justification of the test strategy has been made. Some gaps exist within the test cases, but not to the extent that the broad nature of the test design strategy has been left intact.
70-79	Test cases are provided systematically around a well-selected, well-justified test design strategy. Minimal gaps within test coverage may exist, but no modification should be required for the steps or implementation code
80 +	Test cases are provided in a format that could be implemented without any modification to any aspect of the code. They are exhaustive, insightful, and represent current best practice. All decisions regarding strategies and test data are fully justified.

White box testing (15%)

50	A range of valid test cases are provided that offer a broad understanding of the testing strategy, and valid insight is offered into the selection of the test cases as well as the strategy as a whole. Flaws within pseudocode are permitted at this level, provided that the test strategy's appropriateness can still be assessed.

51-59	An appropriate testing strategy has been followed, with test cases that are, on the whole, in keeping with the test strategy. Justification of the test strategy and selection of pseudocode snippets are consistently relevant to the scenario system.
60-69	Test cases adhere to a well-selected test design strategy. Accurate justification of the test strategy has been made. Some gaps exist within the test cases or within the assumptions behind the pseudocode, but not to the extent that the broad nature of the test design strategy has been left intact.
70-79	Test cases are provided systematically around a well-selected, well-justified test design strategy. Minimal gaps within test coverage may exist, or peripheral branches have not been fully modelled with pseudocode, although pseudocode is broadly exhaustive of the selected non-trivial feature(s).
80 +	Test cases are exhaustive, insightful, and represent current best practice. All decisions regarding strategies and test data are fully justified. No reasonably foreseeable gaps exist within either the pseudocode or the test cases.

Quadrant 3 tests (15%)

50	In addition to the description of the test, some resources have been outlined that would be necessary to implement the test, and some insight is offered as to the reasons why the test would be necessary.
51-59	One or more valid quadrant three tests are described, although some clarification would be needed in terms of implementation details. Resources are identified that would be necessary to the implementation of the test, and a rationale is provided as to the selection of the test.
60-69	Multiple valid quadrant three tests are described. While some clarification would be needed in terms of implementation details, insight is offered that goes beyond the overarching approach. Most resources are identified, and a rationale is provided as to the selection of the tests.
70-79	Multiple valid quadrant three tests are described, with only minor oversight in terms of implementation detail – a competent third party would still be able to implement the tests. Resources are fully itemised, and a sound rationale is

	provided, which includes some discussion of shortcomings of the approaches selected.
80+	Multiple valid quadrant three tests are described in full, providing sufficient detail and clarity as to be actionable by a competent third party with access to the appropriate resources. The resources themselves are fully itemised, with any decisions fully justified. An insightful rationale is provided, with shortcomings meaningfully addressed.

Quadrant 4 tests (15%)

50	A valid quadrant four test has been described in a way that offers sufficient understanding of its role within the scenario system. In addition to the description of the test, some resources have been outlined that would be necessary to implement the test, and some insight is offered as to the reasons why the test would be necessary.
51-59	One or more valid quadrant four tests are described, although some clarification would be needed in terms of implementation details. Resources are identified that would be necessary to the implementation of the test, and a rationale is provided as to the selection of the test.
60-69	Multiple valid quadrant four tests are described. While some clarification would be needed in terms of implementation details, insight is offered that goes beyond the overarching approach. Most resources are identified, and a rationale is provided as to the selection of the tests.
70-79	Multiple valid quadrant four tests are described, with only minor oversight in terms of implementation detail – a competent third party would still be able to implement the tests. Resources are fully itemised, and a sound rationale is provided, which includes some discussion of shortcomings of the approaches selected.
80 +	Multiple valid quadrant four tests are described in full, providing sufficient detail and clarity as to be actionable by a competent third party with access to the appropriate resources. The resources themselves are fully itemised, with any decisions fully justified. An insightful rationale is provided, with shortcomings meaningfully addressed