IN2013 Object-Oriented Analysis and Design
SkinsRUs On-line Consultation System

Dr Peter T. Popov

Version 1.0

07/10/2019 09:12:00

Document History

1.0 New Document.

Introduction

There is one piece of coursework for IN2013, worth 30% of the total mark for the module. You are asked to analyze a scenario and then develop a set of user requirements and analysis models for the software system described in the scenario.

Please note that although the coursework is released at the start of week 3 the coursework tests your knowledge of topics that will be taught in subsequent weeks. You will not be able to answer all the questions until the end of Week 5, unless you read ahead in the recommended text book to the end of Chapter 12 on Use Case Realization.

You are advised to *start working on the coursework as soon as possible*. Certainly do not wait until the deadline approaches: this coursework requires a fair amount of effort, to be spread out over the whole period until the submission deadline, helping you to exercise on the material taught as you proceed.

The deadline is the end of Week 7, **23:55 on the 10th of November 2019**.

Your work must be submitted electronically in accordance with the guidelines below.

Scenario: SkinsRUs On-line Consultation System (WebSRU)

You've been approached by SkinsRUs, a dermatology practice. After a successful operation for a number of years they have decided to expand by offering a global consultation service on-line. They need a web-application, WebSRU that will allow patients from around the world to seek competent consultation over the Internet by submitting a consultation request, which includes a textual description and an image of an anomalous skin condition.

The patients will be charged a fixed price of £20 per consultation with an introductory offer of £10 for the first consultation. There will be a discount for patients who use the service on a regular basis – every 10th consultation is free of charge.

Patients will be able to login securely using their authentication details and check whether their consultation requests have been seen by a dermatologist and, if so, what advice has been given. If they find the advice ambiguous or unclear, they may write a note asking for further clarification, which is recorded by the system. The system should allow for up to 2 clarification requests per consultation request. A clarification advice by a dermatologist is provided free of charge.

Patients are allowed to cancel an on-line consultation request within 24 hours of submitting it unless a consultation advice has already been given. If cancellation is accepted a refund may be given after deducting 3% of the consultation fee to cover the payment clearance fees. If a consultation request is cancelled, the image held on WebSRU will be destroyed.

All payments and refunds are handled by an external Payment Processor such as PayPal, which will be integrated with WebSRU.

The images uploaded with consultation requests will be held in WebSRU in an encrypted form using an encryption key. To improve security, the SkinsRUs personnel (dermatologists or admin staff) may change the encryption key periodically (on demand). Whenever a key is changed, the personnel should provide the old and the new key. The system will validate the old key and if the validation is successful will use it to decrypt all saved images and then encrypt them using the new key.

The dermatologists should be able to login securely and access patient records with outstanding consultation/clarification requests. The system will automatically decrypt the images a dermatologist is trying to access (using the current encryption key).

The application should maintain on-line patient records that include:

- authentication details such as username and password. These are created by the system when a patient submits their *first* consultation request (by uploading an image and a textual description) and makes a full payment for the consultation. The login details are sent to the patient using an email address, which the patient will be required to provide with their first consultation request. The emails will be sent via a 3rd party email (SMTP) server. The patient is expected to store the login details securely and use them in the future to login to their WebSRU account;
- consultation requests (each including an image and the textual description of the specific skin condition), any subsequent clarification requests related to the consultation request, and the advices provided by dermatologists in response to consultation/clarification requests;
- payments made by the patient (payment date, amount paid, last 4 digits of the card used, transaction ID returned by PayPal. Transaction ID is needed to enact a refund).

WebSRU should also maintain a record of the interaction between the dermatologist(s) and the patients:

- a dermatologist should be able to trace the doctor(s) who have provided the initial consultation advice and responded to any subsequent clarification requests;
- a patient should be able to access the consultation/clarification requests and the respective advices, but the identity of the dermatologists, who have provided the advice will be withheld from the patient.

SkinsRUs would like WebSRU to rank the patient records according to the urgency of any pending requests by the patients. If a consultation/clarification request has not been dealt with within 24 hours, the system will raise the patient record priority to *urgent*. Once an urgent record has been dealt with and all pending consultation/clarification requests submitted by the patient have been resolved, the system will restore the patient's record priority to *normal*.

Assignment

You are expected to develop a set of requirements and UML models on WebSRU software and answer the following questions.

Question 1: User requirements

Using the Volere Template, introduced in Lecture 1, specify 1 functional and 1 non-functional requirements for the WebSRU using the provided scenario.

(5 marks)

Question 2: Use case Diagram

Draw a use case diagram for the WebSRU, which covers the functionality described in the scenario. The diagram should include:

- The Actors (primary and secondary) of the WebSRU system. Consider the users of the online consultation service and the external systems, which WebSRU relies on;
- The use cases, which capture the main services provided by WebSRU to the respective users.
- The generalization relationships between Actors.
- The relationships between the use cases (<<include>>, <<extend>> and generalization).

(20 marks)

Question 3: Use case Specifications

One of the central features of WebSRU is that it allows a member of the public to submit a consultation request, i.e. upload an image and a textual description of the skin anomaly. Let's assume that the use case, which serves this purpose, is called "SubmitConsultationRequest", which may have <<include>> and/or <<extend>> relationships with other use cases included in your use case diagram.

Using the template described in lecture 2 of the module, write a detailed use case specification for the use case "SubmitConsultationRequest". Make sure that this specification captures the functionality outlined in the scenario including the *important alternative flows* and is consistent with the use case diagram, i.e. if in the use case diagram the use case "SubmitConsultationRequest" has relationships with other use cases, you should show these relationships in the specification of the use case "SubmitConsultationRequest", as well.

Submitting specifications for the use cases which have <<include>> or << extend>> relationships with "SubmitConsultationRequest", however, is *not required*. [1]

(10 marks)

Question 4: Analysis class diagram

a) Develop an analysis class diagram for WebSRU. Concentrate on the ***problem domain classes***, show their attributes and important operations and the associations between classes.
   - There is no need to include type information, get and set methods, or constructors.
   - Consider a minimal set of boundary and control classes that might be needed for the realization of the use case "SubmitConsultationRequest" (as required in Q5).
   - Relationships:

[1] Such additional use cases specifications, however, may be needed to answer Q5. I would like to encourage you to develop them despite the fact that submitting these is NOT required.

- Use associations in your class models and label them with association or role names, as appropriate, show association directions, and multiplicities, but don't worry about navigability.
- Use generalization (inheritance) between classes where appropriate.
- Don't bother with dependency relationships

(30 marks)

b) ***Substantiate*** your answer by demonstrating competence with the taught techniques for identifying analysis classes and their relationships as follows:

b.1. Apply the noun/verb analysis to the following fragment from the provided scenario:

> "The dermatologists should be able to login securely and access patient records with outstanding consultation/clarification requests. The system will automatically decrypt the images a dermatologist is trying to access (using the current encryption key)".

b.2. Demonstrate the use of CRC cards technique on *3 classes from the problem domain*, e.g. Patient, Consultation and Payment.

b.3. Apply Robustness analysis to the analysis class model by adding to the class diagram a sufficient number of control and boundary classes and checking if the associations between the classes in the class diagram satisfy the robustness analysis rules.

(10 marks)

Question 5: Use case realisation (sequence diagram)

Draw a sequence diagram that realizes the use case "SubmitConsultationRequest". The diagram should cover all possible branches (extensions, alternative flows and if-else's) and loops as defined in your answer to Q3. Make sure that your sequence diagram is consistent with the class diagram developed in Q4 and, of course, with the use case specification developed in Q3.

> ***Hint:*** Note that as a result of developing the sequence diagram your class diagram may change – you may need to add new operations to some of the classes or even add new classes to the class diagram. The use case specifications may change, too: you may discover that the flows (the main and/or the alternatives) may need to be modified.
> The models include in the submission ***must be consistent***. The simplest way to achieve consistency is to complete *all models*, and only then take a snapshot of the class diagram, the use case model (diagram and specification) and of the sequence diagram and include these in your CW submission.

(25 marks)

Submission guidelines

1) Submissions can **only** be made **electronically** via **Moodle**, using the coursework submission area for the IN2013 module.

2) The **deadline** for submission is **23.55 on Sunday, the 10th of November 2019**.

3) Moodle will adhere to the cut-off date/time and will automatically prevent you from attempting to submit your work after the deadline.

4) I suggest that you do not even attempt to work right up to the deadline and instead recommend that you get your submission in well before the cut-off time. The last thing you need is the stress and worry of watching the clock tick and then encountering a problem that delays you. It can and does happen!

Late submission policy

In accordance with the usual policy on coursework submission, **late submission will receive 0%** unless the extenuating circumstances panel approves a request that you submit within the **deadline**, supported by appropriate **evidence**.

Feedback

We will aim to get coursework marked and returned to you by the end of Week 10. General feedback will be provided during the revision lecture in Week 11, and you will also receive individual feedback and a provisional mark via the grade book in Moodle.

Format

All UML diagrams must be created using a UML tool. Diagrams drawn without a tool will be penalized by ***deducting 20%*** from the awarded mark. Take advantage of tools like Visual Paradigm to ensure that your various diagrams are *consistent*.

Your coursework must be submitted in two parts:

1. A ***single PDF file*** with your answers to all 5 questions. Make sure that all your diagrams are exported from the UML tool as images and inserted in a single document, from which you create a single pdf file.
2. A ***Visual Paradigm project***, which contains all your diagrams.

You may also generate separate files with the diagrams, e.g. as pdf files, and submit these as well. The submission area allows you to submit up to 5 files.

UML Diagrams

UML tools are available on PCs in the laboratories, and you can also download a copy for your personal machine (Windows, Mac, or Linux) by following the instructions provided on Moodle.

Note that you may come across variations in UML syntax on websites and within certain textbooks. This is of no consequence for the purposes of this coursework.