SWEN90007 Software Design and Architecture

Diagnostic Imaging Enterprise System

Evan Cranney [*ecranney*]
Shalitha Weerakoon Karunatilleke [*sweerakoon*]

[BitBucket] https://bitbucket.org/{username}/swen90007-diagnostic-imaging-enterprise-system/

SYSTEM OVERVIEW

The purpose of the *Diagnostic Imaging Enterprise System (DIES)* is to support medium to large medical imaging practices in the provision of imaging services.

Background Information

Medical Imaging Practices: "Practices" are groups of

- Medical Specialists (Radiologists);
- Technicians (Radiographers, Sonographers); and
- Support Staff (Administrators, IT staff)

that work together under single business entity (typically a corporation or partnership). Being businesses in themselves, practices must also fulfill all the requisite support functions that this entails: human resources, bookings management, IT support, etc.

Services Provided: The services provided by imaging practices are the

- Taking of medical images of patients (e.g. x-rays, magnetic resonance images, and ultrasound); and
- Reading/reporting on these images (e.g. for evidence of malignant cancer).

Typical Workflow: Practices typically receive patients "by-referral"; this is when an external doctor requests particular imaging services on behalf of a patient. Patients then come into the practice premises to have their images taken. Each image is then read by at least two radiologists, each of whom writes an associated report (stating, for example, whether they believe there is evidence of a cancer). If the reports are in agreement, then they are returned to the patient and referring doctor. If the reports are in disagreement, then a third radiologist (an "arbiter") is required to make a third read of the image. In this case, the arbiter's report is returned to the patient and referring doctor.

Diagnostic Imaging Enterprise System

Features Overview: The features of the *Diagnostic Imaging Enterprise System* are aligned with the primary functions of each of the roles at imaging practices (medical specialist, technician, support staff). These are as follows:

- *Personnel Management.* Information systems staff add and remove staff members' user accounts from the system, associated with particular roles and associated access rights.
- *Bookings Management* ***(Feature A)***. Administrative staff create new bookings for patients after receiving referrals from external doctors. They may also view, update and delete existing bookings.
- *Imaging.* Technicians take images of patients using specialist medical equipment, and upload these to the system.

- *Reporting* ***(Feature B)****.* Radiologists read images on the system and write specialist medical reports on those images. If conflicting reports are written on the same image, then an arbiter (also a radiologist) is able to review those reports.

Feature Summary (CRUD)				
	A: Booking	B: Reporting	C: Imaging	D: Personnel
Technician	-	-	C	-
Specialist	-	CRUD	-	-
Support	CRUD	-	-	CRUD

Properties: DIES satisfies the definition of "enterprise system" in that it has the following properties:

- *Persistent Data.* The data stored (e.g. bookings history, medical images, and reports) on images must be persistent since these must be accessible by different users and over independent sessions. Medical images and reports, by law, must be stored for decades.
- *A Lot of Data.* DIES supports "a lot of of data" in two senses: firstly, the volume of bookings made, images uploaded, and reports written is large (medium-sized practices will take the images of approximately 200 patients daily); secondly, medical images are highly memory-consuming data objects (e.g. a full-body MRI requires approximately 1GB).
- *Concurrent Access.* The system also supports concurrent access. Multiple users are able to interact with the system simultaneously, and - in many cases - in ways that operate on the same data objects. Two radiologist may, for example, be simultaneously reporting on the same image. Likewise, a technician may be uploading an image for a patient simultaneous to a radiologist reporting on a separate image for the same patient.
- *Many User Interfaces.* There are multiple user interfaces for the system; these depend on the function and role of the given user. Administrative staff are presented with bookings interfaces, technicians with an image-uploading interfaces, and specialists with an image-viewing and reporting interfaces.
- *Built on Business Logic.* Features provided by DIES are designed to support the natural workflow of medical imaging practices (booking, imaging, reporting), and must adhere to their business rules. For example, the system should not permit reports to be returned to referring doctors unless the associated image has been read and reported on by at least two (and in some cases, three) specialists. Additionally, the system should not permit updates or deletions of reports that have already been forwarded to the relevant patient and referring doctor.
- *Integrates with Other Enterprise Systems.* DIES must integrate seamlessly with other enterprise systems. These are, namely, accounting systems (e.g. MYOB) that support

invoicing and payroll; external patient health records (e.g. Australia’s My Health Record System); and Medicare.

Feature A: Manage Bookings

The system provides substantial functionality for managing patient bookings. This includes creating new bookings for patients, as well as viewing, updating (e.g. changing the datetime), and removing existing bookings. In support of these functions, the system also performs validation to ensure that new bookings are made on valid dates (e.g. not on dates already passed) and only when a technician is available to take the images. Bookings may also be marked as “completed” or “missed”.

Feature B: Reporting on Images

System functionality also supports reporting on medical images. Technicians upload patient images as they are taking them (a separate feature, *Imaging*). Subsequently, radiologists may access these images remotely to read them. During reads, radiologists are able to highlight important parts of images and write their reports. Other radiologists are then able to access and review these reports. Provided that reports have not already been sent to a referring doctor, it should also be possible to update and delete reports.

FEATURE A: USE CASES

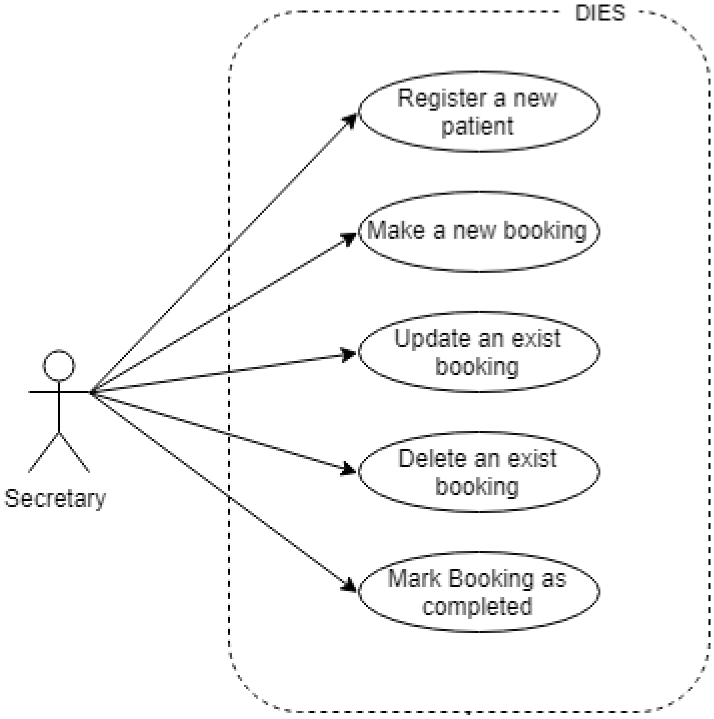

Figure 1: Feature A (Manage Bookings)

A1. MAKE A NEW BOOKING

Primary Actor: Secretary

Stakeholders: Patient

Preconditions: Referral received from an external doctor for patient; secretary has authorisation for bookings management

Postconditions: New booking added to bookings table

Main Success Scenario:

1. Secretary opens interface for making new booking
2. Secretary types patients details (name, phone number and/or email, address)
3. System performs validation over details input
4. System retrieves patient from patients table using details input
5. System calculates available datetimes, based on existing bookings and available technicians
6. Secretary selects available datetime
7. System calculates available technicians, based on selected datetime
8. Secretary selects an available technician for booking
9. Secretary creates booking with specified details
10. System adds new booking to bookings table

11. System shows booking confirmation to secretary

Alternate Scenarios:
3 alt: Secretary enters invalid patient details (e.g. special characters, invalid phone number, invalid address)

- System displays warning message (invalid input)
- Go to Step 2.

3 alt: Secretary does not enter required field.

- System displays warning message (required field not filled)
- Go to Step 2.

4 alt: Patient does not exist in patients table.

- Go to Use Case A5 Step 3.
- Go to Step 5.

A2. UPDATE AN EXISTING BOOKING

Primary Actor: Secretary

Stakeholders: Patient

Preconditions: Booking exists for patient; secretary has authorisation for bookings management

Postconditions: Booking updated for patient with new specified details

Main Success Scenario:

1. Secretary opens interface for searching bookings
2. Secretary searches for existing booking, based on patient name and/or datetime
3. Secretary opens interface for viewing details of existing booking
4. System calculates available datetimes, based on existing bookings
5. Secretary selects new datetime for booking
6. System calculates available technicians, based on selected datetime
7. Secretary selects an available technician for booking
8. Secretary updates booking
9. System updates booking details in bookings table
10. System shows booking confirmation to secretary

Alternate Scenarios: N/A

A3. DELETE AN EXISTING BOOKING

Primary Actor: Secretary

Stakeholders: Patient

Preconditions: Booking exists in bookings table; current datetime precedes booking datetime; secretary has authorisation for bookings management

Postconditions: Booking no longer exists in bookings table

Main Success Scenario:

1. Secretary opens interface for searching bookings
2. Secretary searches for existing booking, based on patient name and/or datetime
3. Secretary opens interface for viewing details of existing booking
4. Secretary deletes booking
5. System requests confirmation for deletion
6. Secretary confirms deletion
7. System removes booking from bookings table

Alternate Scenarios: N/A

A4. MARK A BOOKING AS "COMPLETED"

Primary Actor: Secretary

Stakeholders: Patient

Preconditions: Booking exists in bookings table; secretary has authorisation for bookings management

Postconditions: Booking marked as completed or missed in bookings table

Main Success Scenario:

1. Secretary opens interface for searching bookings
2. Secretary searches for existing booking, based on patient name and/or datetime
3. Secretary opens interface for viewing details of existing booking
4. Secretary marks booking as completed
5. System requests confirmation for marking booking as completed
6. Secretary confirms update
7. System marks booking as completed in bookings table

Alternate Scenarios:

4 alt: Patient missed booking:

- Secretary marks booking as missed
- System requests confirmation for marking booking as missed
- Secretary confirms update
- System marks booking as missed in bookings table
- (*optional*) Go to Use Case A1, create new booking for patient

A5. REGISTER A NEW PATIENT

Primary Actor: Secretary

Stakeholders: Patient

Preconditions: Patient does not exist in patients table; secretary has authorisation to alter patients table

Postconditions: New patient added to the patient table

Main Success Scenario:

1. Secretary opens a interface for patient registration
2. Secretary types patients details (name, phone number and/or email, address)
3. System performs validation over details input
4. Secretary submit the registration form
5. System add new patient to the patient table
6. System shows patient registration confirmation to secretary

Alternate Scenarios:

2 alt: Secretary enters invalid patient details (e.g. special characters, invalid phone number, invalid address)

- System displays warning message (invalid input)
- Go to Step 2.

2 alt: Secretary does not enter required field.

- System displays warning message (required field not filled)
- Go to Step 2.

FEATURE B: USE CASES

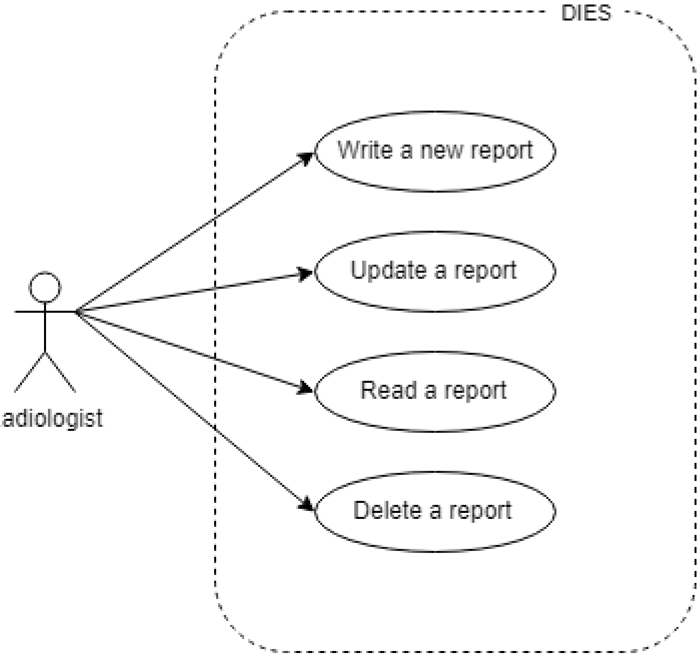

Figure 2: Feature B (Reporting on Images)

B1. WRITE A NEW REPORT

Primary Actor: Radiologist

Stakeholders: Patient, Referring Doctor

Preconditions: Images uploaded to system; radiologist has authorisation for viewing images and managing reports

Postconditions: Report associated with image added to reports table

Main Success Scenario:

1. Radiologist opens interface for searching for images
2. System displays list of image (by patient name and date taken)
3. Radiologist selects an image
4. System retrieves image using metadata
5. System loads interface for viewing image and writing reports, renders selected image
6. Radiologist analyses image
7. Radiologist writes report based on analysis
8. Radiologist highlights relevant sections of image in accordance with report
9. Radiologist submits report
10. System saves report to reports table
11. System saves highlights to highlights table, associated with image

Alternate Scenarios: N/A

B2. UPDATE AN EXISTING REPORT

Primary Actor: Radiologist

Stakeholders: Patient, Referring Doctor

Preconditions: Report exists in reports table; radiologist has authorisation for viewing images and managing reports

Postconditions: Report updated in reports table

Main Success Scenario:

1. Radiologist opens interface for searching for images
2. System displays list of image (by patient name and date taken) and associated reports (by radiologist name)
3. Radiologist selects a report, based on image and radiologist name
4. System retrieves image and report
5. System loads interface for viewing image and writing reports, renders selected image
6. Radiologist updates report
7. Radiologist updates highlights in accordance with report
8. Radiologist submits report
9. System updates report in reports table
10. System replaces highlights in highlights table with new highlights, associated with image

Alternate Scenarios: N/A

B3. READ AN EXISTING REPORT

Primary Actor: Arbiter (Radiologist)

Stakeholders: Report Author (Radiologist), Patient, Referring Doctor

Preconditions: Report exists in reports table; arbiter has authorisation for viewing images and managing reports

Postconditions: N/A

Main Success Scenario:

1. Arbiter opens image
2. Arbiter reads report(s) associated with images
3. Arbiter selects a report, based on image and radiologist name
4. System retrieves image and report
5. System loads interface for viewing image and writing reports, renders selected image

Alternate Scenarios: N/A

B4. DELETE AN EXISTING REPORT

Primary Actor: Radiologist

Stakeholders: N/A

Preconditions: Report exists in reports table; radiologist has authorisation for viewing images and managing reports

Postconditions: Report removed from reports table

Main Success Scenario:

1. Radiologist opens interface for searching for images
2. System displays list of image (by patient name and date taken) and associated reports (by radiologist name)
3. Radiologist selects a report, based on image and radiologist name
4. System retrieves image and report
5. System loads interface for viewing image and writing reports, renders selected image
6. Radiologist deletes report
7. System requests confirmation for report deletion
8. Radiologist confirms
9. System removes report from reports table

Alternate Scenarios: N/A