

4M21 Software Engineering and Design: 2017/2018 Solutions

Elena Punsakaya

15 January 2018

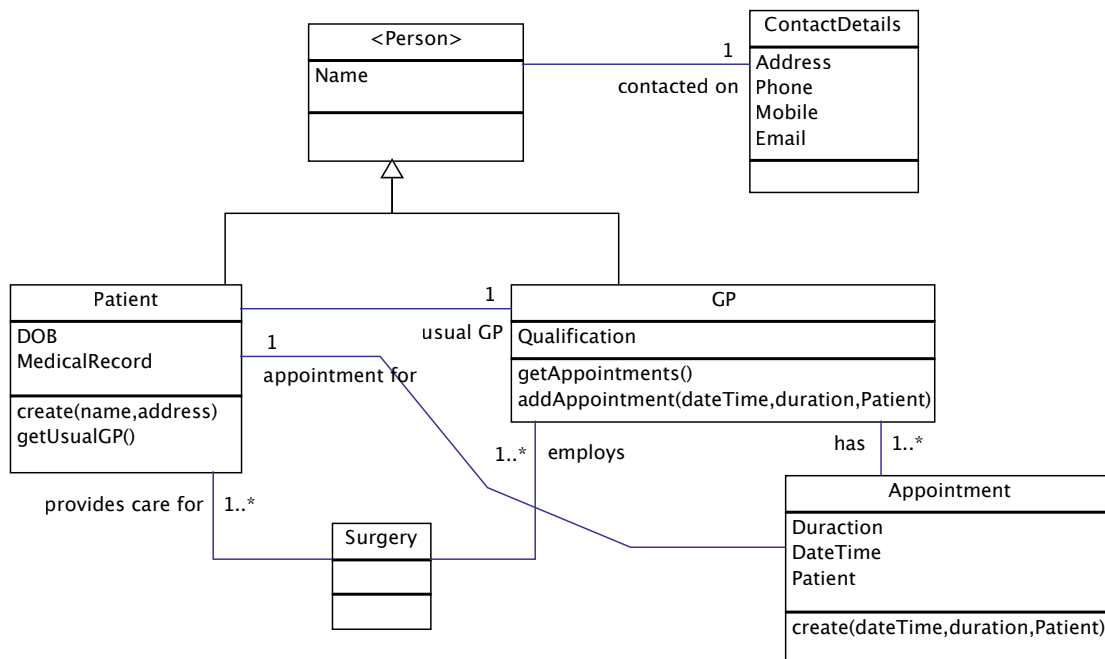
Q1. (a) Provide a definition for *Class Diagram*. Explain why a *Class Diagram* alone is not sufficient to describe a system and identify another type of diagram that can be useful. [10%]

Bookwork.

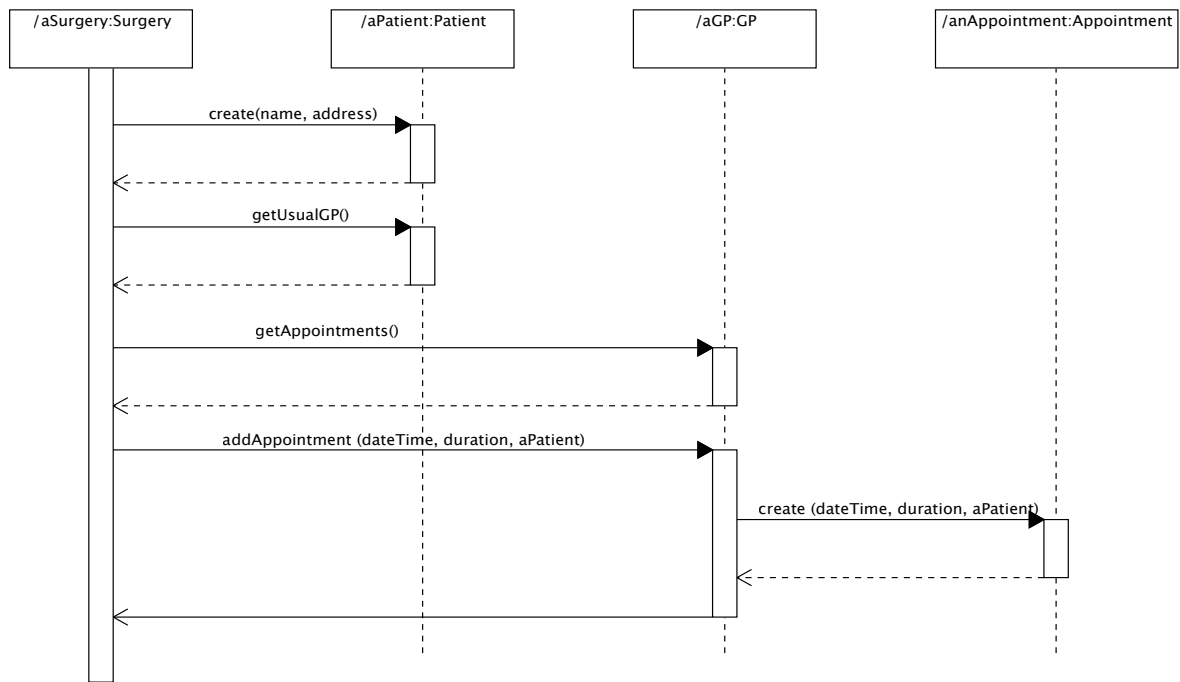
(b) A surgery has a computer system that maintains a list of general practitioners (GPs) working in the surgery and patients it provides care for. The system keeps contact details of GPs and patients and stores patients' personal information such as date of birth and medical records as well as GPs' qualifications.

The system allows surgery staff to schedule appointments for patients to see their GP. The receptionist can access the system by looking up the patient (assume that the name and address identifies the patient uniquely), establishing their allocated GP and checking the list of current GP's appointments. Once a suitable time slot is identified, the receptionist can add a new appointment to the system.

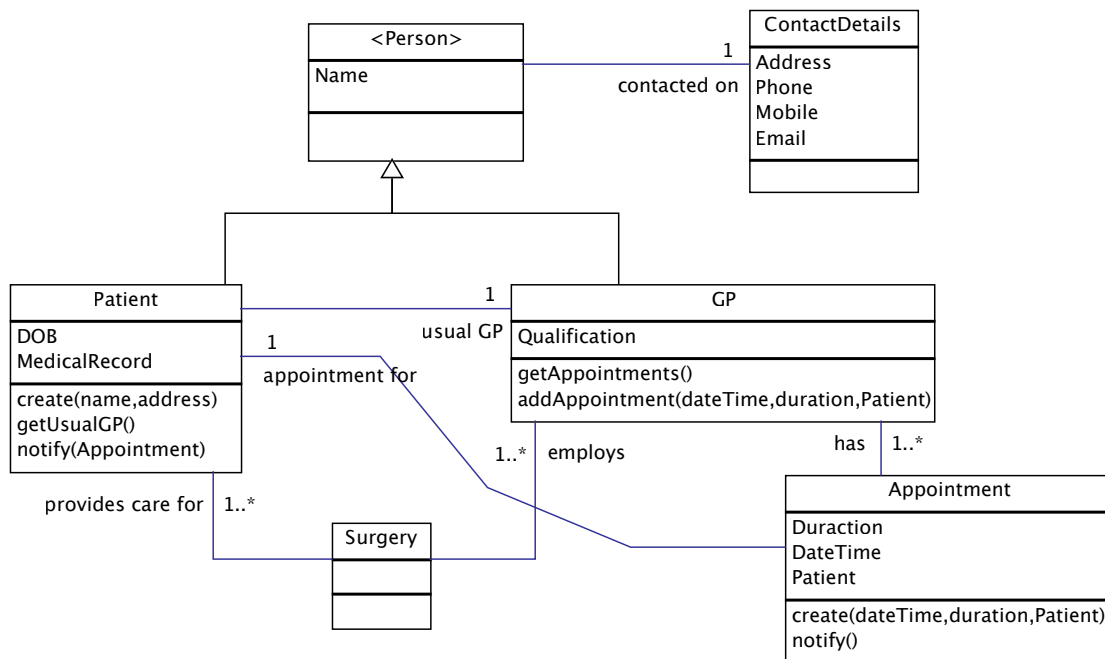
(i) Design the system described above. Provide an illustration of the design with the help of a *Class Diagram*. [30%]

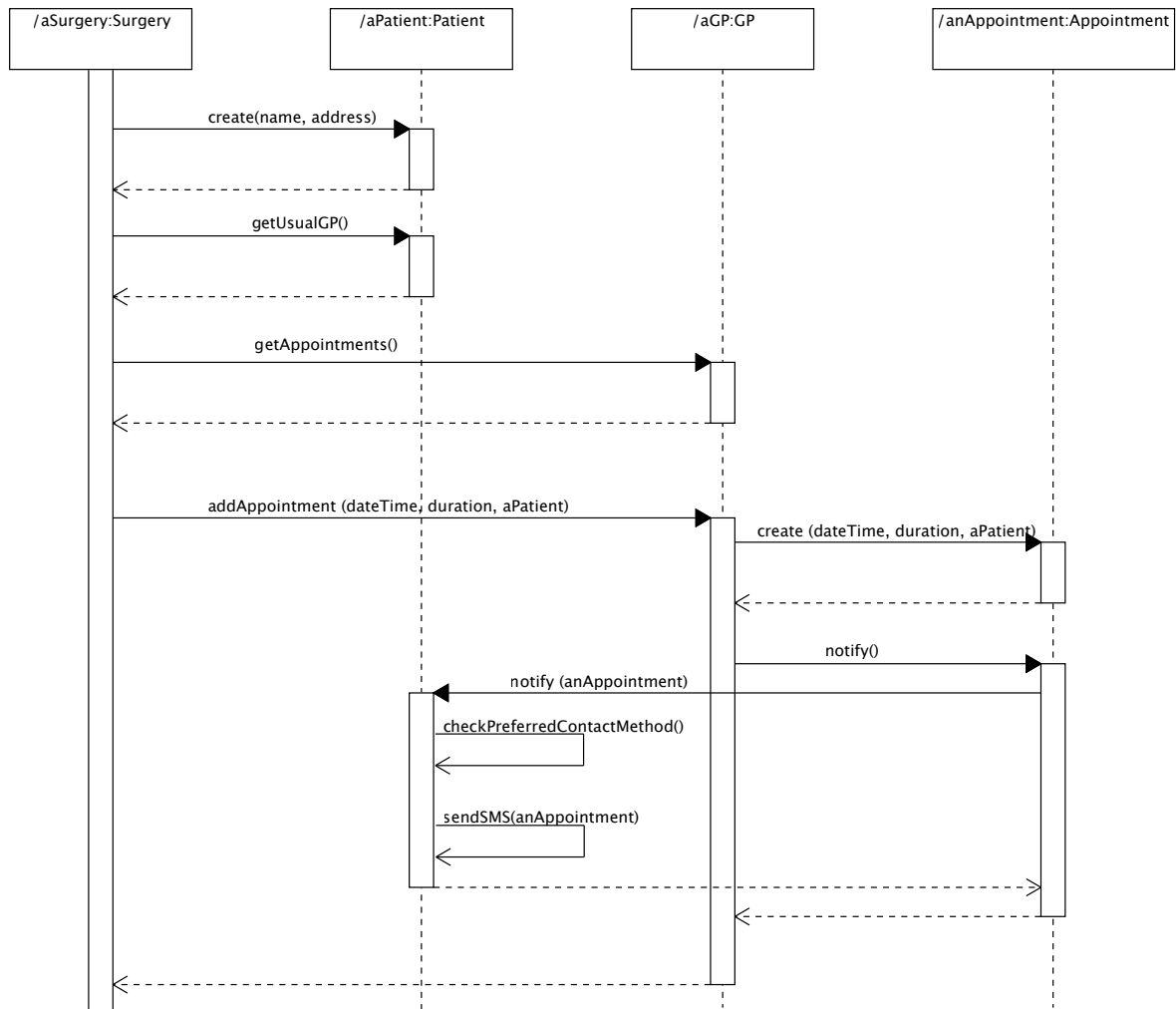


(ii) Draw a *Sequence Diagram* to demonstrate a scenario for a receptionist using the system to make an appointment for a patient. [30%]



(iii) Extend your design to allow the system to send a notification to the patient using their preferred notification method (a text message or email) once the appointment is scheduled. Update the *Class* and *Sequence diagrams* to illustrate this extension. [30%]





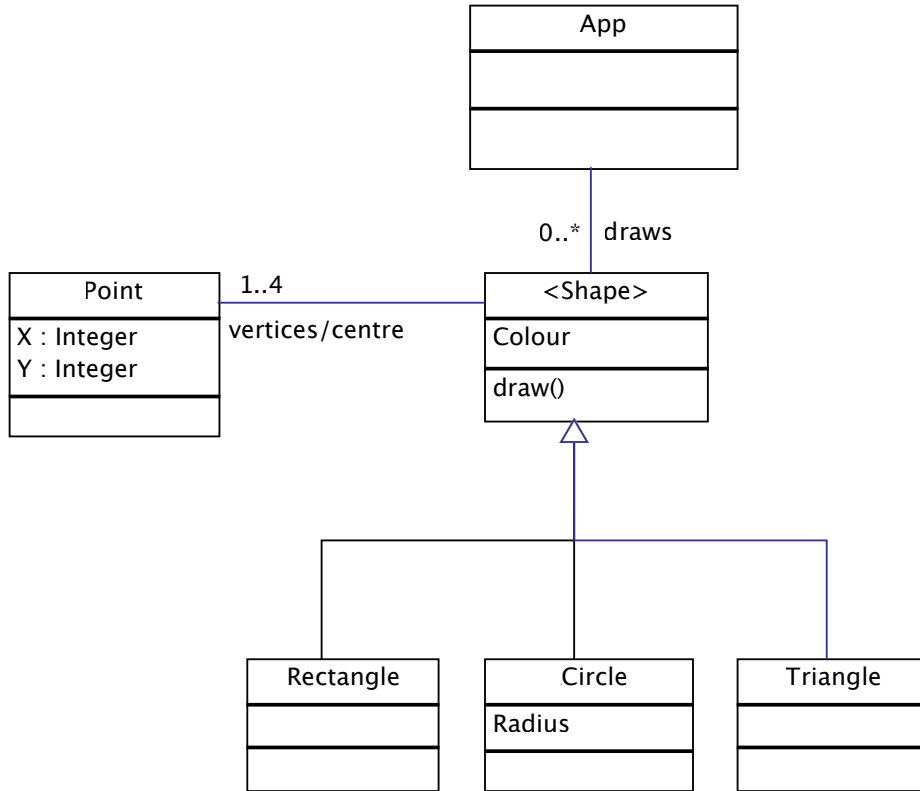
Q2. (a) Provide the definition of *Refactoring*. [10%]

Bookwork

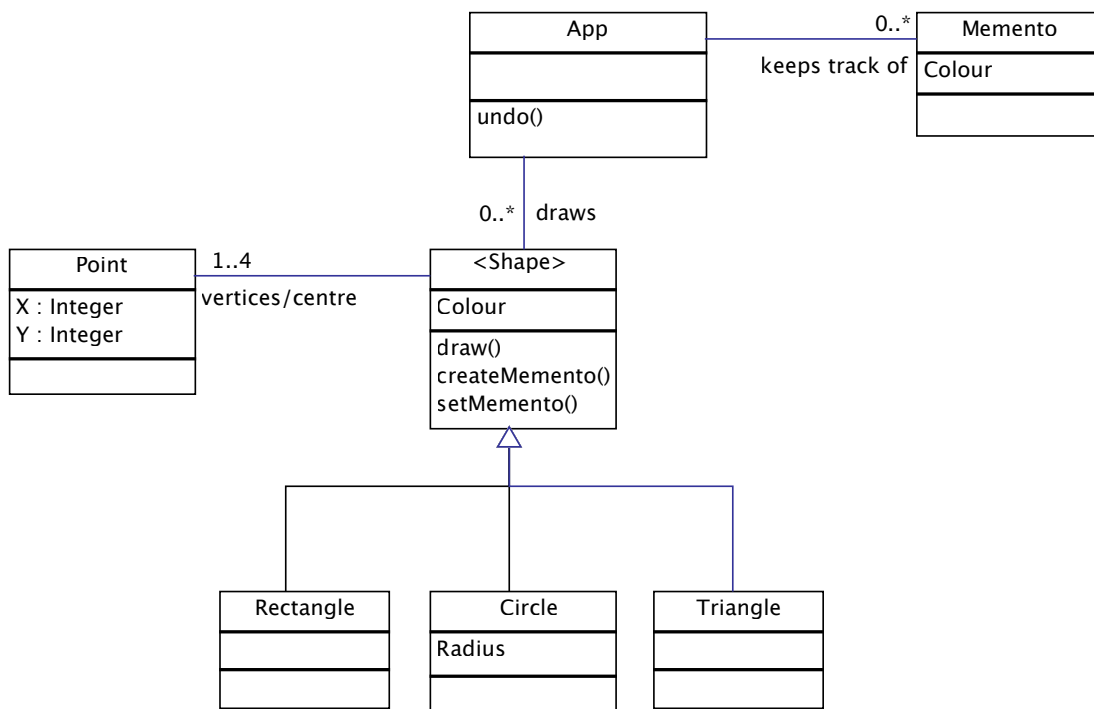
(b) A simple mobile application allows the user to draw primitive shapes: rectangles, triangles and circles. Once drawn, the user can tap on the shape to select it and choose its colour. The *Class diagram* in Fig. 1 shows the main classes..

(i) Review the design and using *Refactoring* techniques improve it by generalising shape classes. [30%]

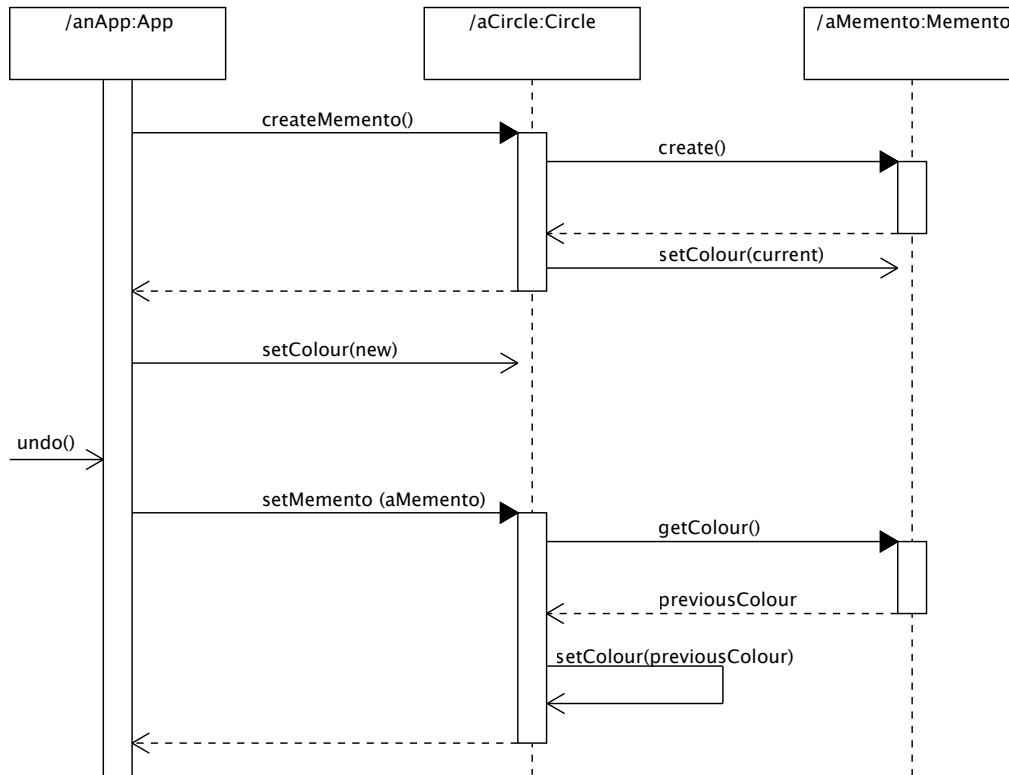
One of the possible solutions is presented below.



(ii) Identify a Design Pattern to add the Undo functionality for changing the colour of a shape. Draw the Class diagram for the updated design. [30%]



(iii) Draw the Sequence diagram to demonstrate changing the colour and then using the Undo functionality to revert it. [30%]



Q3..

- (a) Describe the purpose of Activity Based Planning in User Interface design, provide some examples and explain why it can be useful. [10%]
- (b) A specialised online video streaming company maintains a website for children and youth to create and share animated videos. The company would like to create an application that customers can download on their tablet and use to discover and watch some outstanding animated videos. The users should be able to search videos by keywords and watch them online.

(i) Design the *User Interface* (UI) for a software application for a portable tablet device with a touchscreen that implements this functionality. Identify all main screens and interaction elements and explain their purpose and design constraints. [30%]

(ii) Extend the functionality of the application by allowing the users to go back to recently watched videos, mark some of the videos as their Favourites and download some of them to their tablet. The users should be able to see all animations added to their Favourites. Identify any additional screens required to support these features. [40%]

(iii) Having analysed the app statistics, it has been discovered that a large percentage of users downloaded the app on their smartphone instead of the tablet. Optimise the design of the application to improve the user experience on smartphone devices. [20%]

- a) Activity based planning involves analysing the requirements and designing user interface by analysing and listing typical user activities. These activities motivate the design.

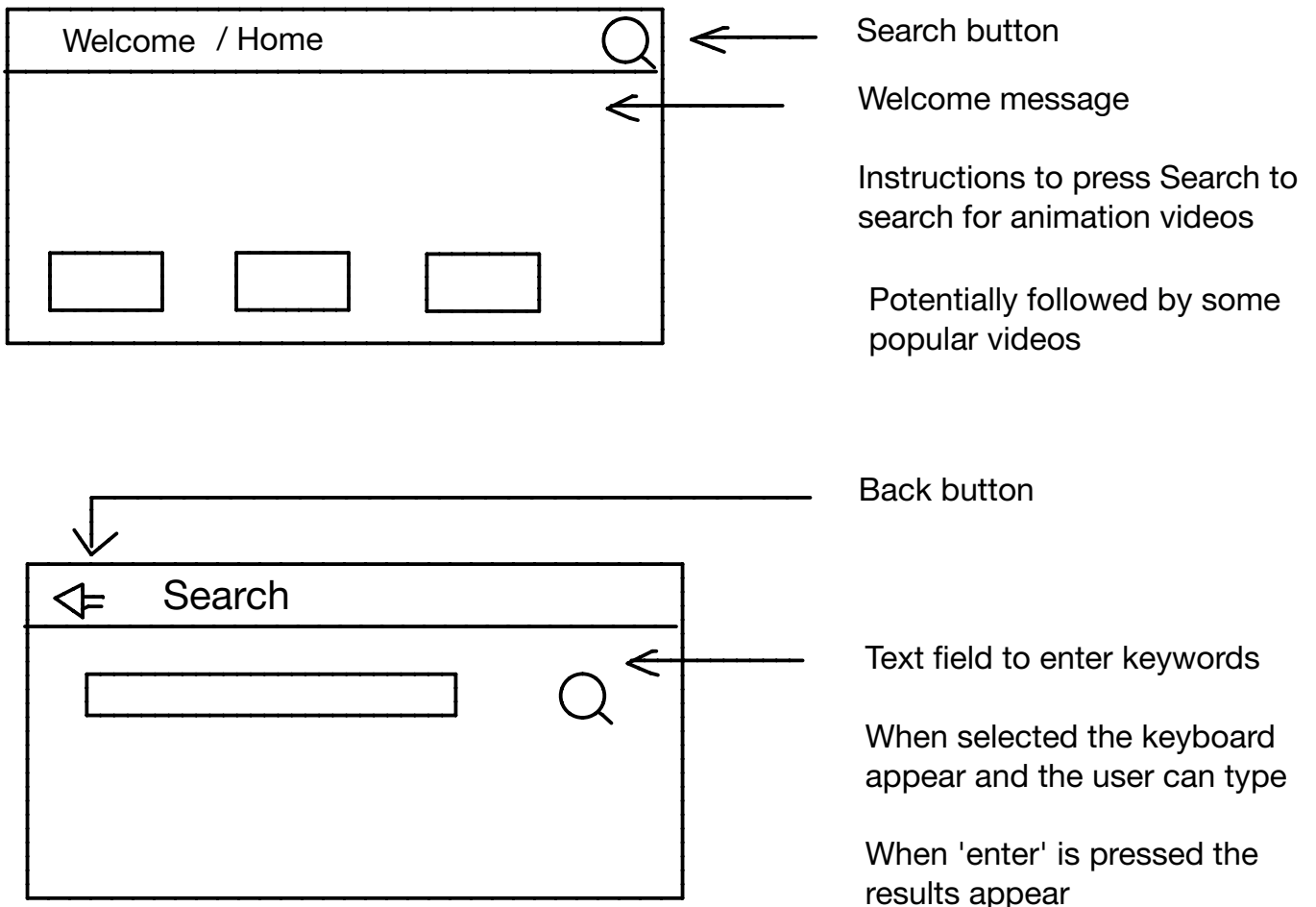
These activities also provide a basis for testing.

A use case describes an activity (a typical user activity) from start to end. It consists of

- definition of user's goal
- list of pre-conditions (information needed before the task can start)
- criteria for successful completion of the task
- list of main steps in the activity flow
- any extensions/alternatives to this use case

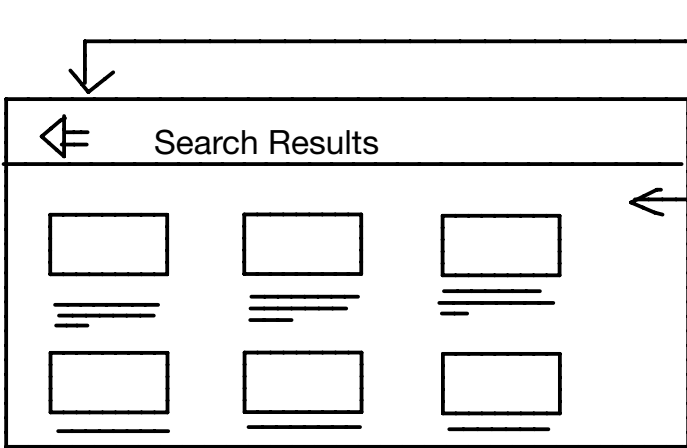
b)

- i) One of many possible solutions is presented below.



For all network connectivity dependent actions a progress indicator is shown and an error screen may appear if the connection fails.

Help button can be introduced in the right top corner.

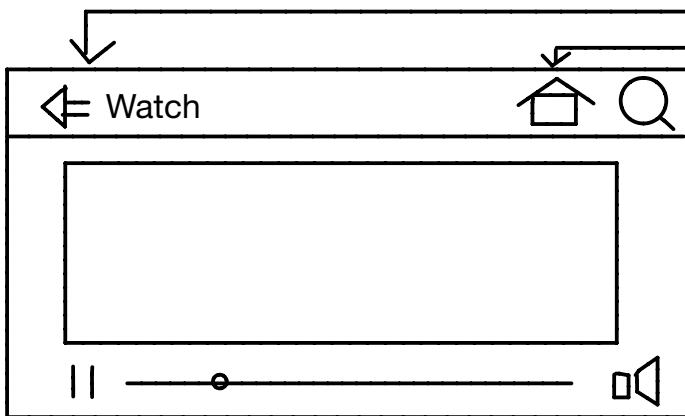


Back button takes the user to the home screen

Search results appear with some images followed by brief text description

Scroll view is enabled if required

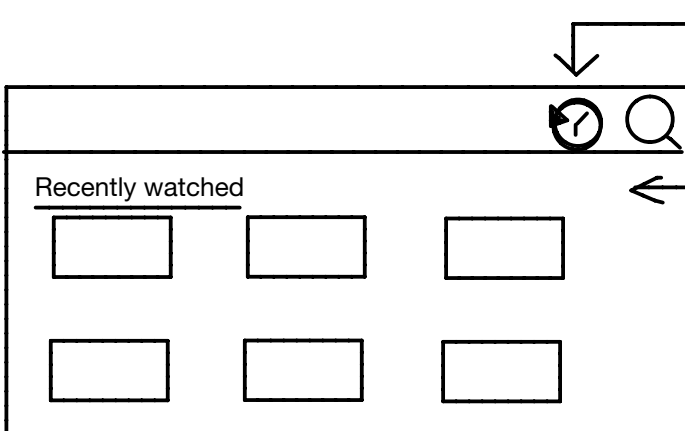
Users can click on the images to watch the video



Back button takes the user back to search screen

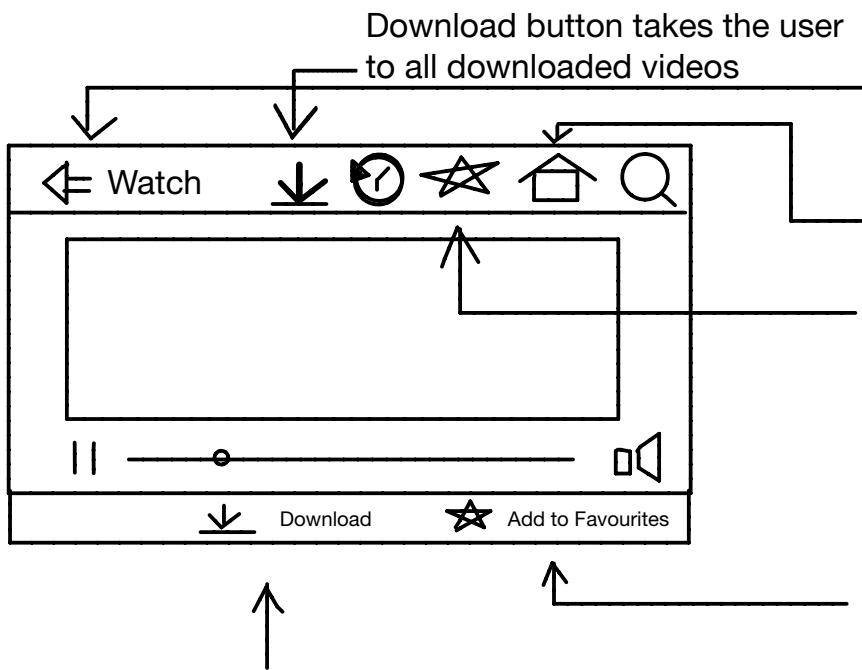
Home button takes the user to home screen

- b)
ii) One of many possible solutions is presented below.



Introduce Watch Again button

Alternatively/additionally provide a link to/display Recently watched videos on the home screen



Back button takes the user back to search screen

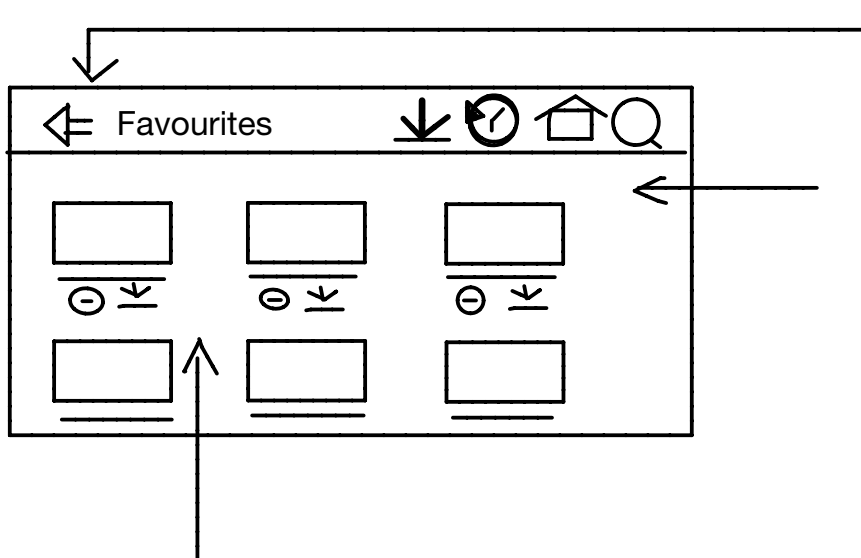
Home button takes the user to home screen

Favourites button takes the user to the screen where all videos previously added to Favourites are displayed.

Add to Favourites button is added. Pressing on this button adds the corresponding video to the list of Favourites followed by a brief confirmation messages confirming to the user that the video was added to Favourites.

If the video is already in Favourites Remove from Favourites button appears instead. ⊖

Download button is added. Pressing on this button initiates download of the video to the user's device. If there is not enough space, there is no internet connection or there are any other problems corresponding error is displayed. Otherwise, progress bar is shown. If already downloaded Remove from Device option appears instead. ✕



Back button takes the user back

Favourites appear as images followed by brief text description

Scroll view is enabled if required

Users can click on the image to watch the video

If Favourites list is empty a messages is displayed informing the user that the list is currently empty and explaining how to add videos to `Favourites.

Remove from Favourites, Download button (or Remove from Device if already downloaded ✕) options appear next to each video. The icons are chosen according to OS system guidelines.

The same options are added to the main Watch Video screen

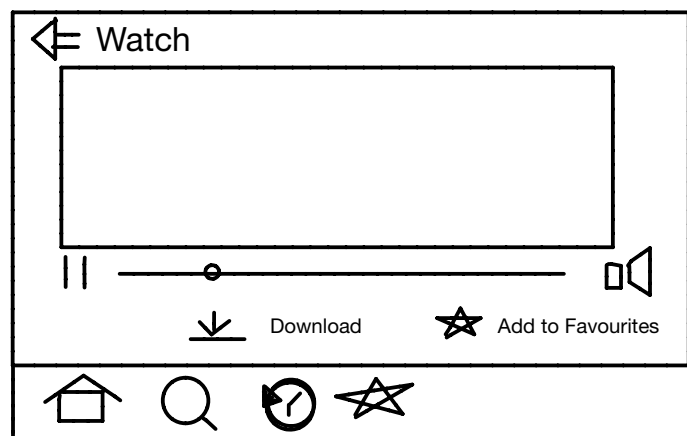
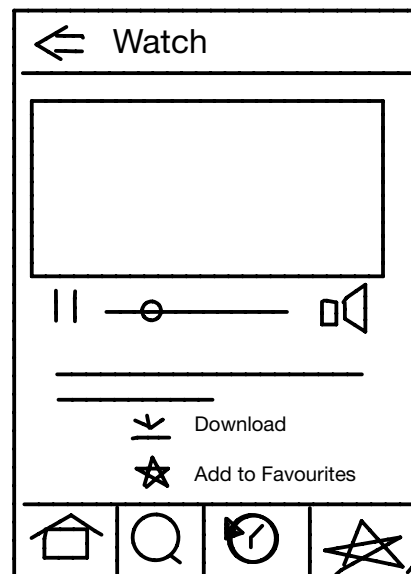
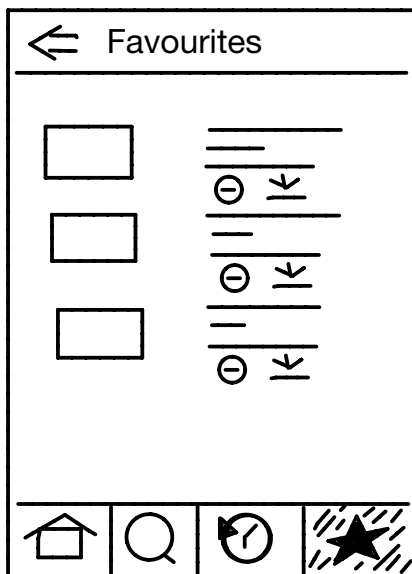
(b)

(iii) One of many possible solutions is presented below.

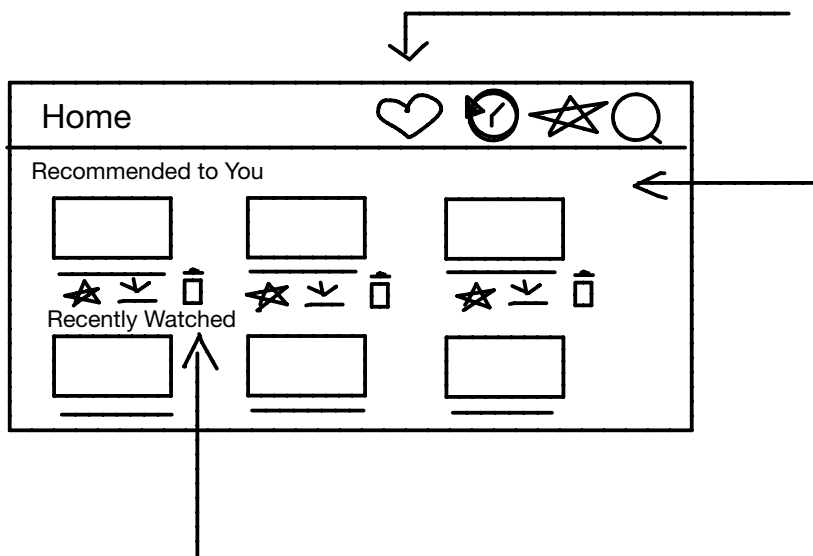
Smartphone touchscreens tend to be smaller than tablet screens, the quality of the screen might be worse, the hardware in general might be more limited, internet connection might be limited. The users tend to use device holding it in one hand. As a result,

- the items should be presented on the screen as a list
- lower resolution images are more appropriate to make sure that they appear quicker
- it's very likely that scrolling has to be enabled
- an option of a lower resolution video might be offered
- both portrait and landscape views should be optimised for a smaller screen and one hand operation, the portrait view should support both browsing and watching
- the buttons needs to be introduced according to OS system guidelines

One of the potential examples of implementation is illustrated below



Extra



Add to Favourites, and Don't show again options appear next to each video. The icons are chosen according to OS system guidelines.

For You button is added, pressing on it takes the user to the page with the Recommended items.

Alternatively/Additionally some of the Recommended to you items are presented on the Home screen, clicking

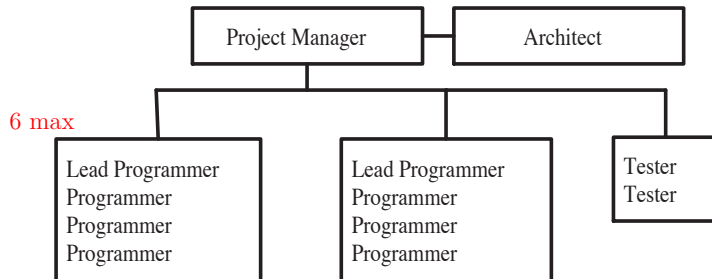
Scroll view is enabled if required

Users can click on the image to watch the video

If Favourites list is empty a messages is displayed informing the user that the list is currently empty and explaining how to add videos to `Favourites.

Q4. A large company specialising in bespoke software development has won a contract to deliver a software tool for monitoring a particular parameter in the operation of a new-generation nuclear power plant.

- (a) Describe the Team that is likely to be working on the project and clearly define their roles and responsibilities.[20%]
-



Architect: The principal designer, defines the overall architecture, module structure and all major interfaces, usually also an expert in the associated technology. Responsible for Specification and High Level Design.

Project Manager: Responsible for scheduling/rescheduling the work, tracking progress and ensuring that all of the process steps are properly completed (on time, on budget).

Lead Programmer: Leader of a programming team. Will typically spend 30% of his/her time managing the rest of the team.

Programmer: Implements specific modules and often implements module test procedures.

Tester: Designs test and validation procedures for the completed software. Tests are based on initial specification and will focus on the overall product, rather than the individual modules.

-
- (b) Specify the Software Model that is likely to be employed for the development of the software, list they key stages in the development process and explain the advantages of this approach for this type of software. [20%]
-

It's a small however potentially safety-critical system which has to work reliably from the first moment it goes live. It is critical to get the system requirements right and dedicate significant amount of time analysing not just the functional requirements but also various "abuse cases" including various types of attacks or natural disaster scenarios, i.e. safety and security requirements. A formal waterfall model is therefore appropriate although depending on the specific requirements

The Waterfall model – sequential design process, progress is flowing steadily downwards through the phases. The key stages are

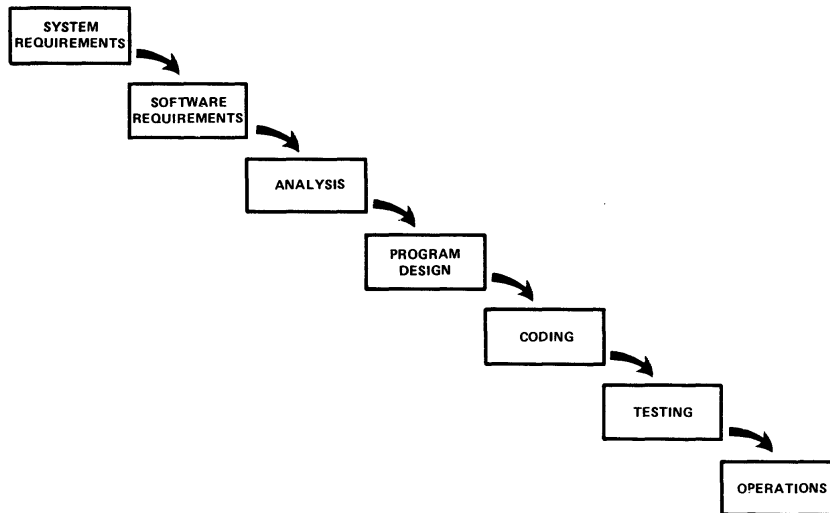


Figure 2. Implementation steps to develop a large computer program for delivery to a customer.

Advantages:

- early clarification of system goals and different types of requirements
- can charge the client for changes to the requirements
- works well with management tools

It is possible and extremely beneficial to spend a lot of time on system specification for this type of software.

- (c) Discuss what could go wrong with the project and what sort of risks the company should be ready to mitigate. [30%]
-

Inconsistent or not well investigated/analysed system requirements, not well understood details are one of the potential failures of the projects.

System requirements might be misunderstood or misinterpreted when they are translated into software requirements, some of the requirements might not be described precisely or analysed thoroughly.

Some of the aspects might not be discovered during the initial analyses stage.

The project might not be completed on time / on budget.

The software might fail to meet functional specification.

The system might not perform in realistic real-world conditions.

The users might not use the system as intended.

Safety or security requirements might not be met.

- (d) Describe the measures that the company could implement to reduce the likelihood of any potential crises and disasters. [30%]
-

It is critical to get all functional, safety, security and other requirements right and a thorough analyses should be conducting when one "is deciding what to build." Expert advice might be required from the specialists in nuclear energy, law enforcement (safety), security engineering, etc.

Effective formal project management tools have to be employed starting from formal waterfall model but evolving if necessary at a later stage into a more iterative process if required.

Effective testing strategy must be design in consultation if needed with specialists in nuclear energy, law enforcement and security engineers; including

- integration testing

- verification and validation
- resource exhaustion, errors, recovery
- performance testing, stress testing and testing under load
- usability testing
- non functional testing

The exam was pleasantly well answered by most candidates.

Q1 The question was designed to test the understanding of the key concepts of the object-oriented design and the ability to apply it in practice. It ended up being a popular question and most students were able to go through an independent design process and communicate its results clearly using the standard notation although some found the question quite challenging. Not everyone was careful when working with both class and sequence diagrams as often sequence diagram did not correspond to the class diagram.

Q2 The question was on understanding and application of refactoring and the use of design patterns in the design process. The question was done reasonably well, however, some students found it difficult to draw a sequence diagram for the identified design pattern possibly due to lack of the detailed understanding of the main principles of this particular pattern.

Q3 This was a user interface design question that was answered well by most of the candidates. A popular question that candidates were able to complete without any major challenges, however, some candidates did not read the questions carefully and omitted some functionality/features.

Q4 A reasonably straightforward, however less popular, question on software engineering methodologies and their application. Those who did attempt the question answered most parts well. Some candidates did not read part c) carefully and did not provide a direct answer to the question.