7 Development

- describe software requirements
- talk about website architecture
- describe programming steps
- discuss future plans and schedules

Requirements analysis

1 Work in pairs. Put these stages of the software development process in the best order.

- ☐ The customer checks and approves the final version.
- ☐ Speak with the people who will use the new software and analyse how they will use it.
- ☐ Plan the project, write the specifications and prepare instructions for the programmers.
- ☐ Test and debug the code.
- ☐ Write the code.

Reading **2** Read this interview from *Computer World*, an online magazine, in which Jess Wong, a systems analyst, talks about her job. Which of the stages in 1 does she mention?

COMPUTER WORLD

CW: So, Jess, could you tell us about the requirements analysis process?

JW: Well, first, we talk to the client to find out who the users will be. Then we interview as many users as we can. This can be difficult because we have to look at every step in the process very carefully, in a lot of detail.

CW: And what's the next step?

JW: Next, we put together the specification document. This shows everything that the program does. And by that I really mean everything! We have to write down what every button does, what you see on every pop-up menu and so on.

CW: Is this where you draw flow charts?

JW: Yes, and the user interface.

CW: And does the client check the specification document?

JW: Certainly. We want the client to look at every part of it, to be sure that they are happy with it. We usually have to make a few changes at this stage but this is usually quite quick. Then we can hand over to the developers so that they can do their part of the job.

3 Read the interview in 2 again and answer these questions.

1 Why does Jess talk to the client at the beginning of the process?
2 Why are the interviews sometimes difficult?
3 What does the specification document contain, besides writing?
4 Why does the analyst want the client to check the specification document?

Vocabulary **4** Complete these definitions with the words in the box. Use the stages in 1 and the interview in 2 to help you.

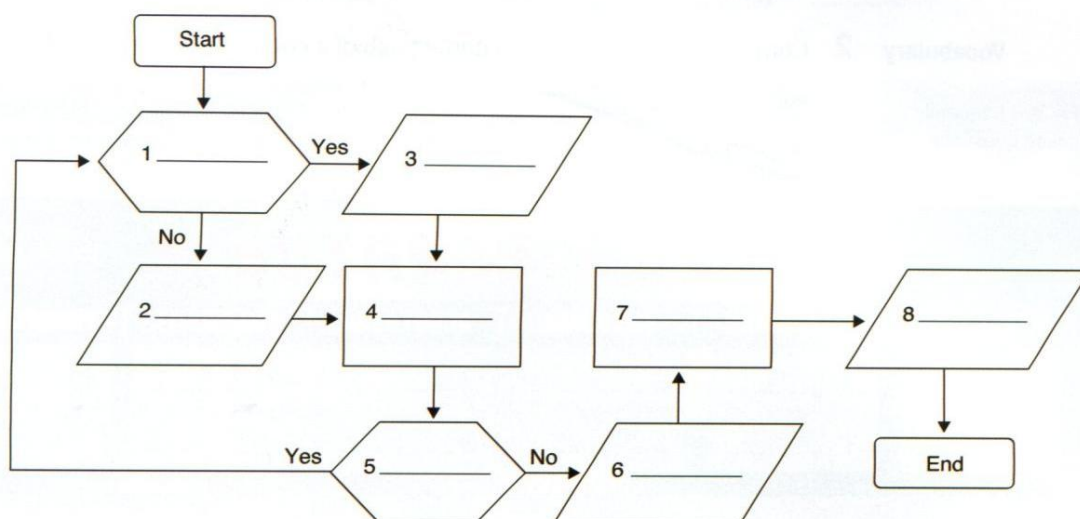| approve	analyse	debug	detail	interview

1 all the separate features and pieces of information about something _____
2 think about something very carefully, step-by-step _____
3 officially say that you are happy with something _____
4 ask someone questions formally in order to find out information _____
5 find problems in a computer program and correct them _____

Listening **5** ▶ **36** Listen to a systems analyst talking to a worker. He is preparing specifications for a pizza shop website to take orders online. Which stage from 1 is he at?

6 Complete 1–4 in this flow chart with the steps in the box. Then listen again and check your answers.

> Ask which type of standard pizza.   Ask which toppings.
> Customer wants standard pizza?   Write order on order sheet.

```
                    ┌─────────┐
                    │  Start  │
                    └────┬────┘
                         ↓
         ┌──────────────────────┐  Yes  ┌──────────────┐
    ┌───→│  1 _____         │──────→│  3 _____   │
    │    └──────────┬───────────┘       └──────┬───────┘
    │              No                          │
    │               ↓                          ↓
    │  ┌─────────────┐   ┌─────────────┐   ┌────────────┐   ┌─────────────┐
    │  │ 2 _____   │──→│ 4 _____   │   │ 7 _____  │──→│ 8 _____   │
    │  └─────────────┘   └──────┬──────┘   └─────┬──────┘   └──────┬──────┘
    │                          ↓               │                  │
    │         Yes  ┌──────────────┐  No  ┌───────────┐       ┌─────────┐
    └─────────────│  5 _____   │─────→│ 6 _____ │       │   End   │
                  └──────────────┘      └───────────┘       └─────────┘
```

7 ▶ **37** Now complete 5–8 in the flow chart in 6 with the steps in the box. Then listen to the second part of the conversation between the systems analyst and the worker and check your answers.

> Ask for delivery address.   Calculate delivery time.
> Customer wants another pizza?   Tell customer delivery time.

Language

Expressing user requirements	
We use *should*, *have to* and *need to* to express **requirements**.	The program **should be** easy to use. It **needs to be** fast but it **doesn't have to look** good.
We can also use *want* + object + infinitive.	The client **wants the program to run** on old versions of Windows.

Speaking **8** Work in pairs. Use the flow chart in 6 to say what the program should do.

First, the program should find out what kind of pizza the customer wants: standard or choice of toppings.

9 Work in small groups. A shop selling fashionable clothing for young people wants an online order system. Think about the software and make a list of requirements.

The website should look interesting for young people. It should show ...

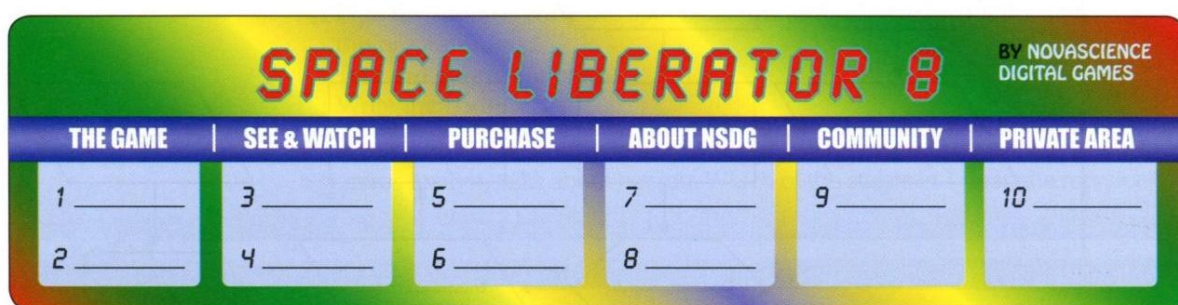**10** Compare your list from 9 with another group's list.

Website design and architecture

1 Work in pairs. Think of a typical website and discuss these questions.

1 Which pages do most websites have (e.g. contact details)?
2 Which of these items, or other items, is often at the top of each page? Which is below that? What else might you find at the top of a web page?
   • contact details   • menus   • search   • title
3 What makes a website easy or difficult to use? What makes it interesting? Think about how easy it is to find things, what the website looks like and what is on it (e.g. photos).

2 Complete the menus on this home page of a computer game website.

> FAQs = frequently asked questions

| Company blog | Contact us | FAQs | How to pay | How to play |
| Images | Login | Players' forum | Prices | Videos |

SPACE LIBERATOR 8
BY NOVASCIENCE DIGITAL GAMES

THE GAME	SEE & WATCH	PURCHASE	ABOUT NSDG	COMMUNITY	PRIVATE AREA
1 _____	3 _____	5 _____	7 _____	9 _____	10 _____
2 _____	4 _____	6 _____	8 _____		

3 Read this case study about a website development project. What do you think a fan site is? Was the project successful? How do you know?

> SEO = search engine optimisation

A web developer's work

Andrea Pinheiro da Silva is a web developer who is well known for the websites she has developed. Her websites have won several awards. Here, she describes a recent project.

'Recently, I worked on a project for SLFan, a fan club for the *Space Liberator* game. There were a few other sites for fans of the same game, so it was important that it looked exciting and dynamic, to get more site visits. The customer also wanted their new site to have two different areas: a public one for general content and a private one with premium content for paying fans.

We decided to use a combination of PHP, HTML5 and MySQL. PHP was chosen to keep costs down (it's open source) and for its flexibility. It's also very versatile: it integrates easily with many other website tools. We mainly used HTML5 for multimedia content but also used other systems so that the site can be viewed on many types of devices, including mobile devices. And MySQL was just right for the private area: forum posts and other private data can be stored in the MySQL database.

Speaking of mobile devices, one of the challenges was to make the site viewable on mobile phones and smaller tablets. We had to make sure that every page can be seen in mobile format. We also had to work on SEO because high search rankings were required. That involved working closely with SLFan.

The end result was a website that the customers were very pleased with. They were great people and real fun to work with!'

Vocabulary **4** Find words in the case study in 3 that match these definitions.

1 accessible by anyone (paragraph 2) _____
2 information, images, video, etc. on a website or in an application (paragraph 2) _____
3 better or more expensive than others (paragraph 2) _____
4 two or more things that are used together (paragraph 3) _____
5 that can be used in many different ways (paragraph 3) _____
6 things that are difficult to do (paragraph 4) _____
7 possible to see (paragraph 4) _____
8 position on a page of search results (paragraph 4) _____

Language

The passive

We use **the passive** when the action is more important than the agent (the person or thing doing the action), or when it isn't necessary to mention the agent at all.	*His website **is linked** to my website.* *The new website **was viewed** many times.* *Videos **can be viewed** on this site.*

5 Rewrite these sentences in the passive.

1 They found a problem.
2 The customer requires a dynamic, exciting website.
3 We used PHP for this website.
4 People can watch videos on this website.
5 People can download useful PDFs from this website.

Speaking **6** Look at this website navigation chart. Which pages are linked to the home page? Which page is the parent node? Which pages are the child nodes? Which are the grandchild nodes?
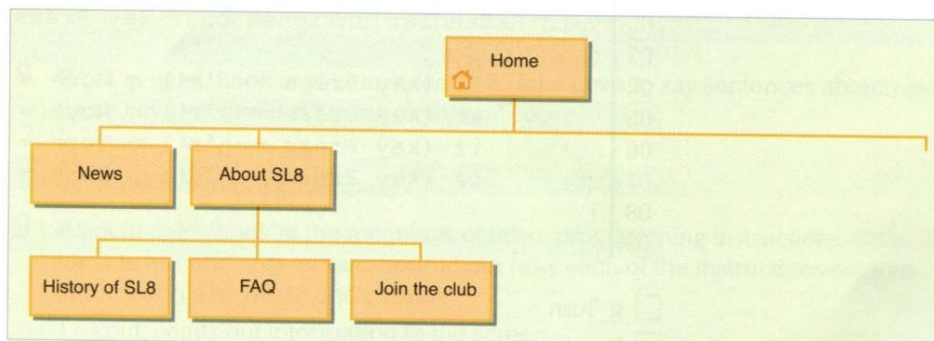
7 Work in pairs. Student A, look at the information on page 68. Student B, look at the information on page 71. Follow the instructions.

Writing **8** Read the text in 3 again. What was done to create the website? Complete this email to your manager. Use the passive.

Hi Natasha,

Just to summarise, this is what we did to create the website:

• The code was written in PHP, JavaScript ...

Software development

1 Work in small groups. Have you ever seen any programming code? Would you like to be good at programming?

2 Work in pairs. Look at the section of code and the explanations and answer these questions.

1 Find an example of a constant in the code.
2 What do you think the value of x is, after the third instruction?

'a' is a variable. A variable is something that can change its value (which can be a number). The opposite is a constant: a constant can't change its value. For example, here the number '3' is a constant but 'a' is a variable: '3' is always '3' but 'a' can have any value an instruction gives it: it can be 1, 2, 3 or any other number. This instruction gives it the value '3', which it keeps until another instruction changes it. Variables can have any name, and sometimes variable names are quite long. For example, 'g_Turn' is a variable name.

		line of code
01	a = 3;	
02	b = 2;	programming instruction to show something on the screen
03	x = a + b;	
04	cout x;	

3 ▶ **38** Listen to the first part of a conversation between two programmers talking about this code, which controls a robot using a mobile phone. Number the variables in the order they are explained.

```
01  int g_Move = 0, g_Turn = 0;
02  void RxHandler(unsigned char key_Press)
03  {
04          if (key_Press == 'a') g_Move = 1;
05          if (key_Press == 'f') g_Move = 2;
06          if (key_Press == 's') g_Turn = 1;
07          if (key_Press == 'd') g_Turn = 2;
08  }
09
```

☐ g_Turn
☐ key_Press
☐ g_Move

4 Complete these sentences. Then listen again and check your answers.
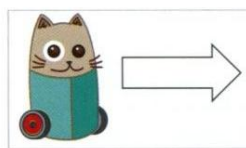
1 If *g_Move* has the value *0*, the robot _____ .
2 If *g_Turn* has the value *0*, the robot _____ .
3 If you press 'x' on the phone, *key_Press* has the value _____ .

5 ▶ **39** Listen to the second part of the conversation between the two programmers. Which lines of code are they talking about?
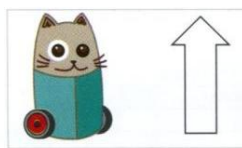
6 Listen again and label these illustrations with the keys that correspond to each direction.
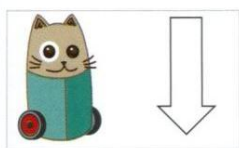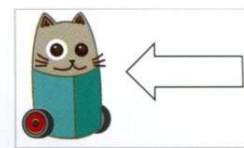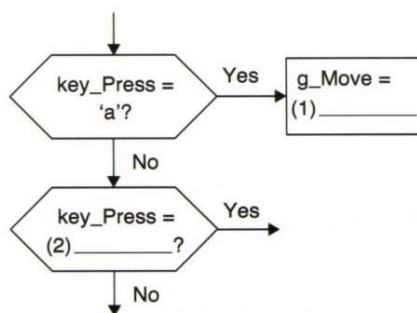

1 _____   2 _____   3 _____   4 _____

Speaking **7** Work in pairs. Look at the code in 3 again and complete this flow chart. Which instructions in the code in 4 do they represent?


```
            ↓
    ⬡ key_Press =  ⬡  Yes  →  ┌─────────────┐
    ⬡    'a'?      ⬡          │ g_Move =    │
                              │ (1)_____ │
          │ No                └─────────────┘
          ↓
    ⬡ key_Press =  ⬡  Yes  →
    ⬡ (2)_____ ?⬡
          │ No
          ↓
```

Language

Make and cause

We use *make* + object + bare infinitive and *cause* + object + *to*-infinitive to explain how something controls something else.	The 'f' key **makes the robot go** backwards. This instruction **causes the computer to print** something.

8 Work in pairs. Look at your answers in 6. Take turns to say sentences about what these key sequences do using *make* and *cause*.

a → f → d → s

'a' makes the robot go forward.

Speaking **9** Work in pairs. Look at the meanings of these programming instructions. Take turns to ask and answer questions about how each of the instructions controls information. Use *make* and *cause*.

1 cout: sends out information to the screen
2 cin: takes input from the keyboard
3 rename: changes a filename
4 exit: stops a program
5 time: returns the number of seconds since midnight on 1 January 1970
6 remove: deletes a file

A: Which instruction makes the computer show something on the screen?
B: 'Cout'.

10 Work in pairs. Take turns to explain to your partner what these items do. Use *make* and *cause*.
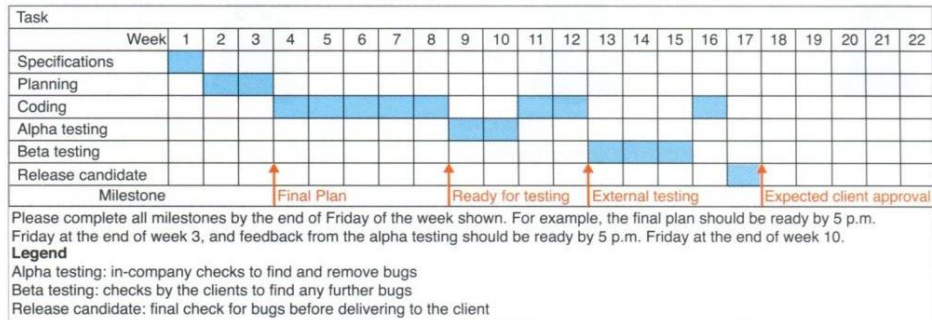
a mouse
A mouse makes the cursor move on the screen.

1 the 'Maximise' button   4 the 'Off' switch
2 a right click   5 the 'Send' button
3 a double click on a file icon   6 the 'Save' button

Project management

1 Look at this Gantt chart. Why might someone use a Gantt chart?

Task																							
Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Specifications	■	■																				
Planning			■																			
Coding				■	■	■	■	■				■				■						
Alpha testing									■	■												
Beta testing													■	■	■							
Release candidate																	■					
Milestone			↑ Final Plan						↑ Ready for testing			↑ External testing					↑ Expected client approval					

Please complete all milestones by the end of Friday of the week shown. For example, the final plan should be ready by 5 p.m. Friday at the end of week 3, and feedback from the alpha testing should be ready by 5 p.m. Friday at the end of week 10.

Legend
Alpha testing: in-company checks to find and remove bugs
Beta testing: checks by the clients to find any further bugs
Release candidate: final check for bugs before delivering to the client

Vocabulary

2 Complete these definitions with the words in the box. Use the Gantt chart in 1 to help you.

> alpha testing   beta testing   coding   feedback   milestone
> release candidate

1 writing software _____
2 an important stage in a project _____
3 the first stage of testing software _____
4 the second stage of testing software _____
5 information about problems or how good something is _____
6 the final version of software, if no big bugs are found _____

Listening

3 ▶ 🎧 40 Listen to a programmer and project manager discussing the schedule for the project in 1. Are they near the beginning or end of the project?

4 Listen again. What was the mistake on the Gantt chart? Correct the chart in 1.

Language

Schedules

We use *plan to*, *be scheduled to* and *be due to* in the present simple to talk about **schedules**. All three phrases are followed by an infinitive.	*We're scheduled to finish* this project on Friday and *we're due to start* the next one on Monday. *We plan to test* the software next week.
We use the present continuous if the event is part of a fixed plan and we can clearly imagine it happening.	Alpha testing *is finishing* next week.

5 Work in pairs. Look at audio script 40 on page 79. Underline all the examples of the language from the Language box.

Pronunciation

6 ▶ 🎧 41 Listen to two short conversations and underline the stressed words in audio script 41 on page 79. Then practise the conversations with a partner.

Speaking

7 Work in pairs. Look at the Gantt chart in 2 and take turns to ask and answer questions about the schedule. Use language from the Language box.

A: *What's scheduled for week 11?*
B: *We're due to start coding again during week 11, after the alpha testing.*

8 Work in pairs. Take turns to ask and answer questions about your work or study schedule.

Business matters

1 Work in small groups. Read this scenario and answer the questions.

> You work for a computer games company. Your manager has asked you to prepare a rough plan for a new project: a website to advertise a new computer game. It will be similar to other game websites but with a special extra feature: an online version of the game that people can try out before buying the real one. This feature will need a lot of development.

1 What is special about the new website compared with others of the same type?
2 What is likely to be the most difficult part of the new website?

2 Work in the same groups. Read the scenario in 1 again and discuss these points. Then draw a site map based on your decisions.

Decide:
1 what pages to have.
2 what type of content to include (e.g. video? any premium content? a private area?).
3 how to navigate the site: which pages should be linked to which other pages?

3 Work in the same groups. Draw a Gantt chart for the project in 1. Add the stages in the box to your chart.

| alpha testing | beta testing | client approvals | coding | planning |
| release candidate | requirements analysis |

Task																					
Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Milestone																					

4 Work in the same groups. Use your Gantt chart from 3 to explain your project to another group. Then compare your charts. What differences are there? What might be the reason for these differences?

Writing **5** Write a short report to your manager describing your project. You can use the information on page 51 to help you. Use these headings in your report:

• Introduction (Give background information about the project.)
• Stages in the project (Describe the stages you planned in 3.)
• Conclusion (Say when you think the project will finish.)