

# Higher Diploma in Science in Computer Science

---

2024-2026





**Rosanne Birney**  
*Database*



**David Drohan**  
*Mobile App*



**Jimmy McGibney**  
*Devops  
Security*



**Siobhan Roche**  
*Programming*



**Caroline Cahill**  
*Computer Systems*



**Richard Frisby**  
*Devops*



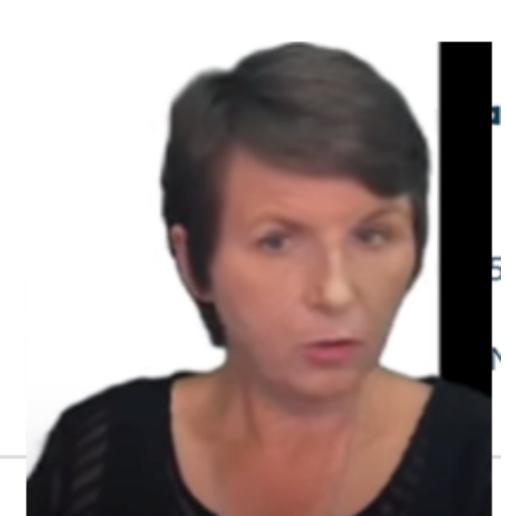
**Laura McGibney**  
*Student  
Engagement*



**Frank Walsh**  
*Computer System  
Full Stack 2*



**Eamonn de Leastar**  
*Full Stack 2*



**Mary Lyng**  
*Database*



**Mairead Meagher**  
*Programming*



**Peter Windle**  
*Programming*



**Colm Dunphy**  
*Project*



**Joan Mangan**  
*Programme &  
Placement  
Coordinator*



**John Rellis**  
*Web Dev 1  
Web Dev 2*



# **Agenda**

1. Context & Objectives

2. Programme Structure

3. Semesters & Modules

4. Calendar, Timetable & Assessment Sequencing

5. Module Summaries

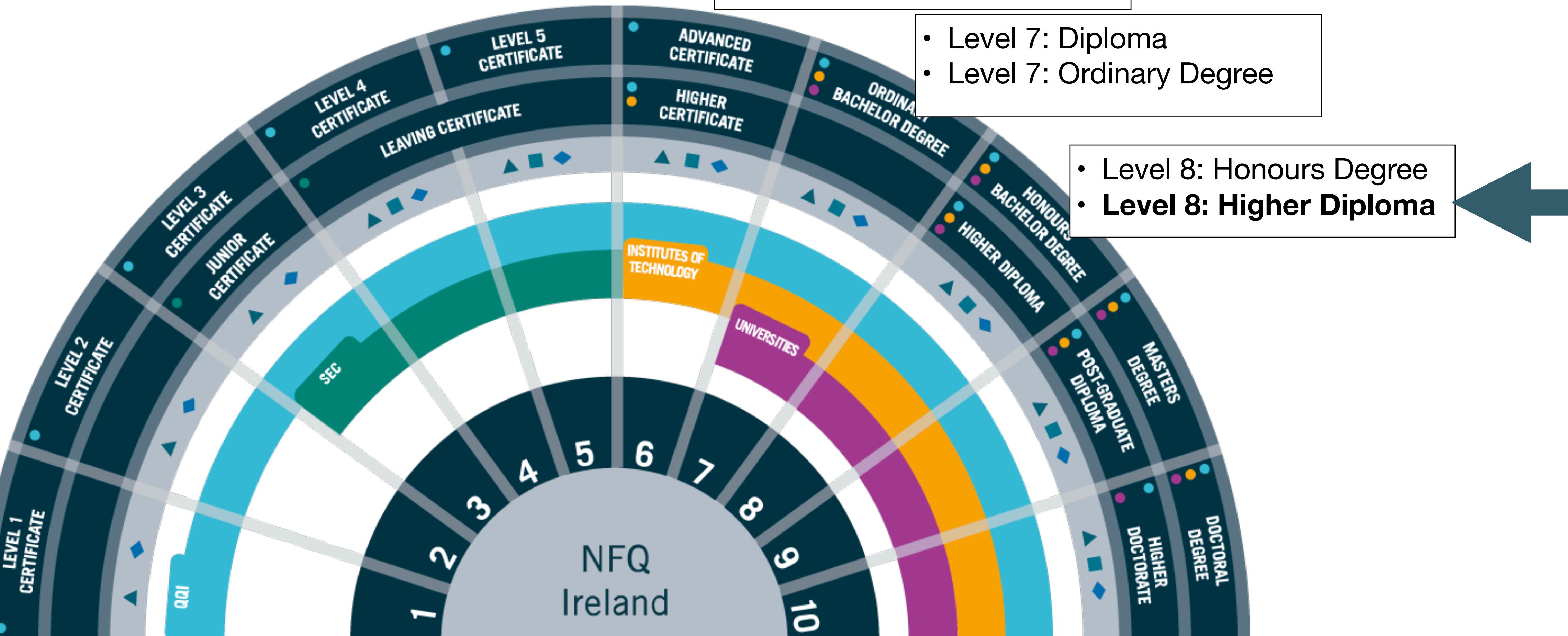
# 1. Context & Objectives

# National Framework of Qualifications / Levels

- Level 6: Certificate
- Level 6: Higher Certificate

- Level 7: Diploma
- Level 7: Ordinary Degree

- Level 8: Honours Degree
- **Level 8: Higher Diploma**



<https://nfq.qqi.ie/>

5

ADVANCED CERTIFICATE	ORDINARY BACHELOR DEGREE	HONOURS BACHELOR DEGREE	MASTERS DEGREE	DOCTORAL DEGREE
HIGHER CERTIFICATE		HIGHER DIPLOMA	POST-GRADUATE DIPLOMA	HIGHER DOCTORATE

# Key Programme **Features**

- Immersion
- Specialisation
- Industry Partnership

# Immersion in Computing Knowledge



*“The participants will be **graduates** who have already obtained significant **transferable skills** by comparison with other undergraduate students...”*

*“Semester 1 participants will undertake a broad immersive set of modules in the **fundamentals** of computing...”*

*“The **pace** of delivery will have to be **significantly higher** than for normal undergraduate programmes...”*

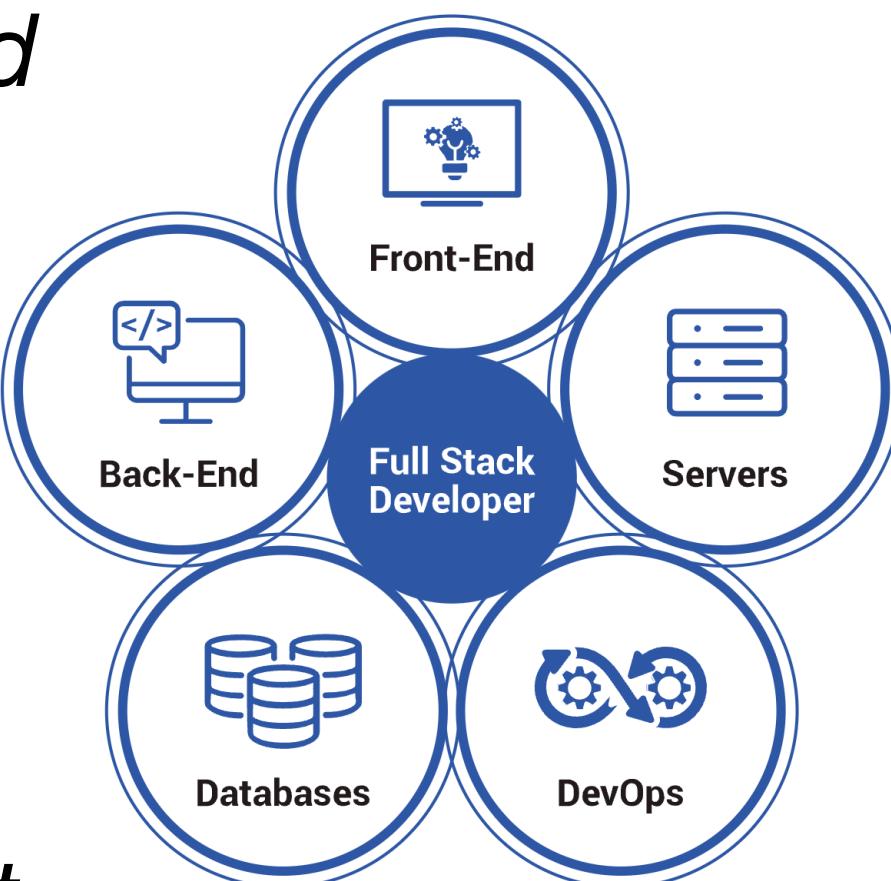
## Deepening and Specialisation



*“In semester 2 ... a **specialisation** which reflects their own strengths as demonstrated on the programme to date...”*

*“.. a focused set of modules and project-work designed to bring candidates quickly to the industry entry standard ...” **Junior Software Developer (Full Stack Oriented)***

*“Participants will be expected to select their specialisation based on their achievement in semester 1 and their own ambitions...”*



# Industry experience and professional development

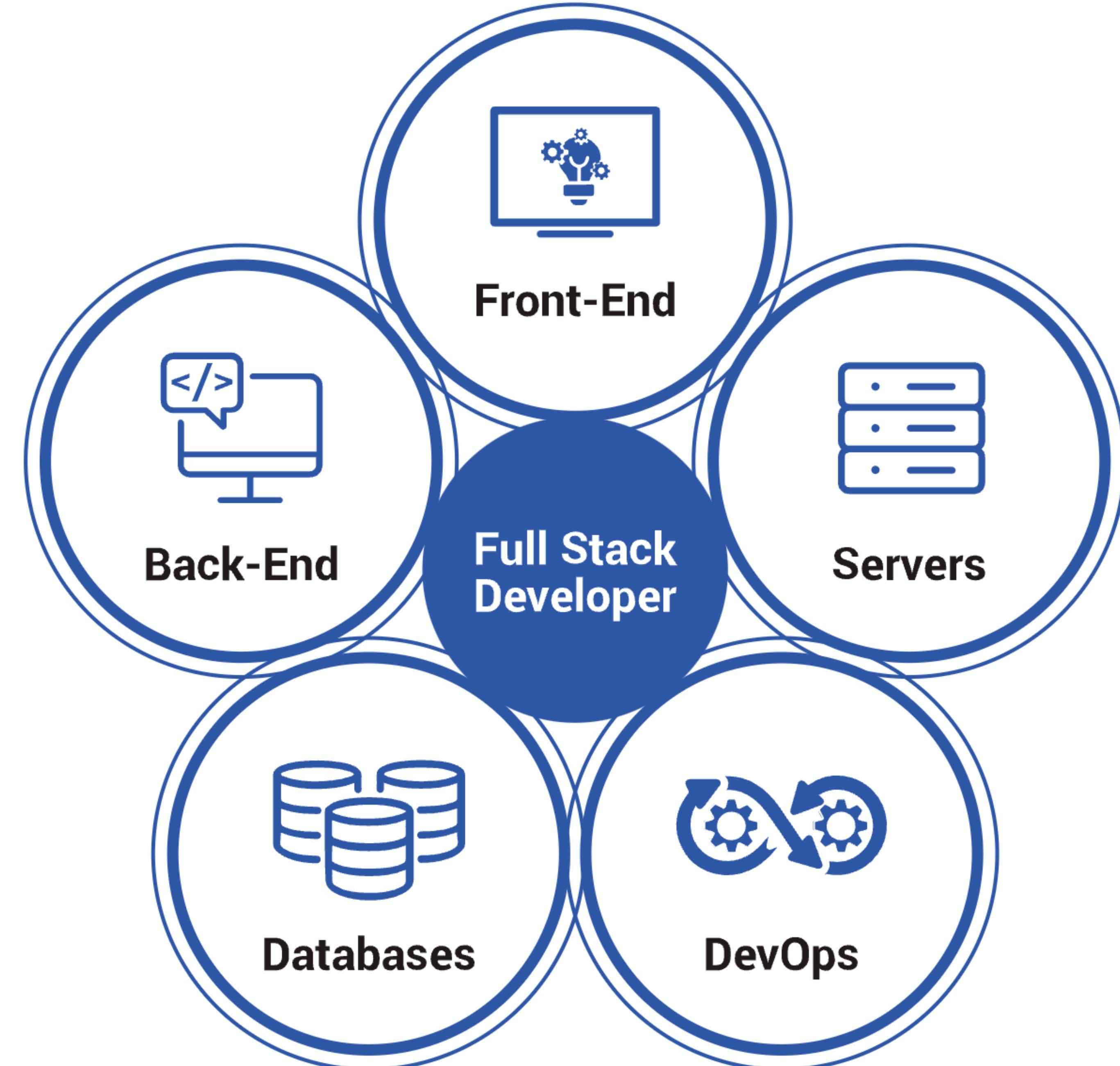


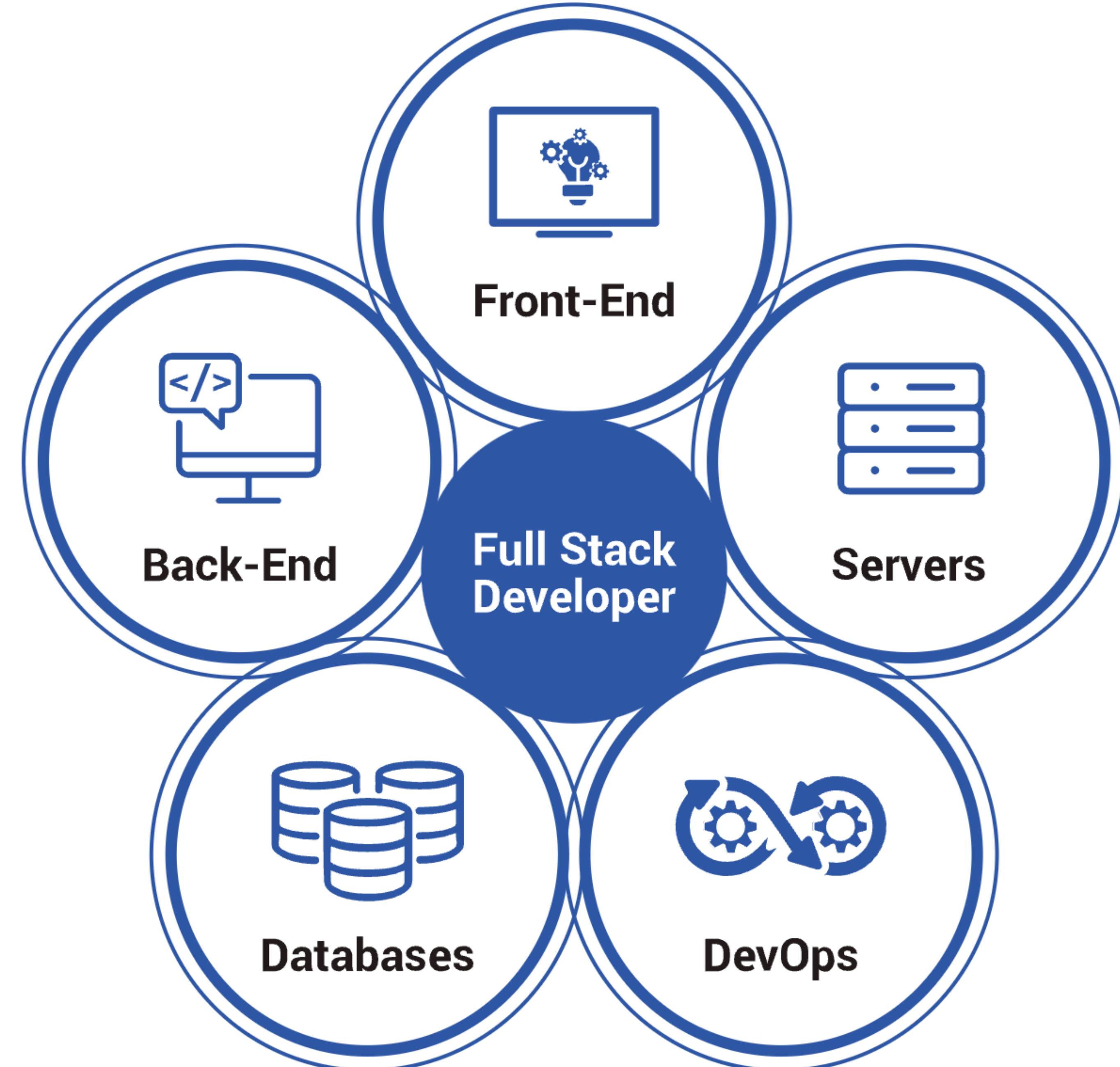
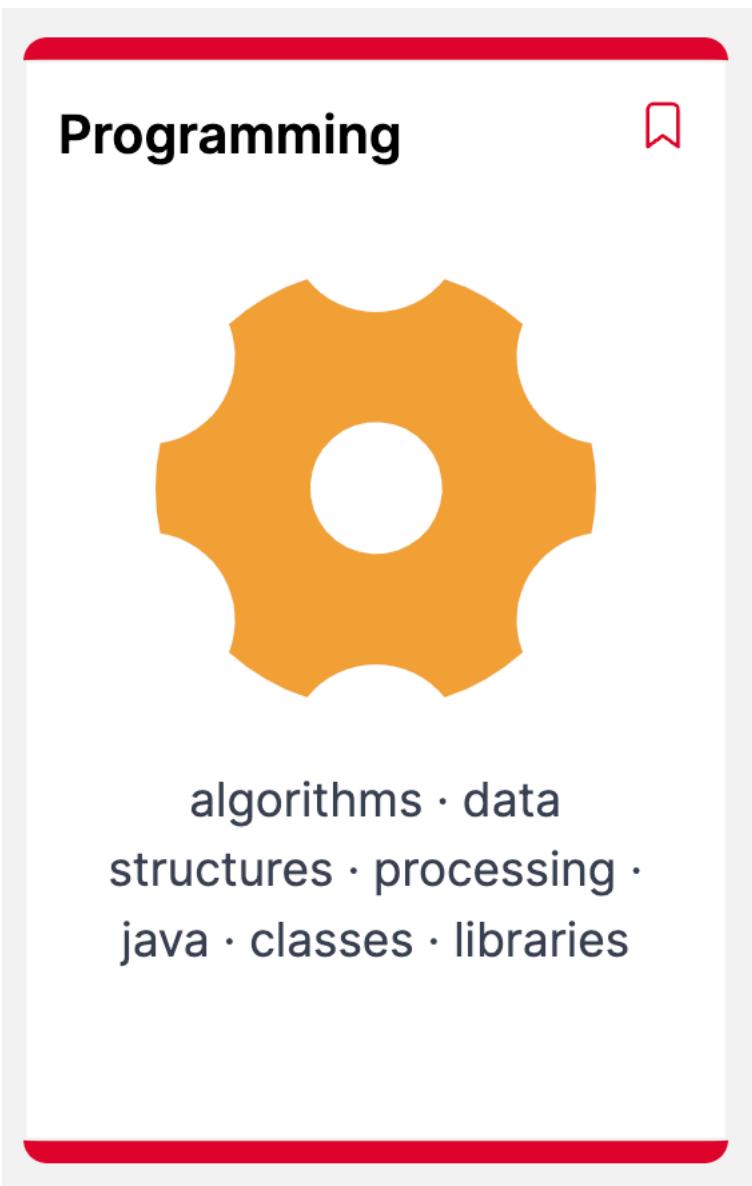
***“Internships or work placements are seen as crucial to providing graduates with the context and confidence in their new knowledge...”***

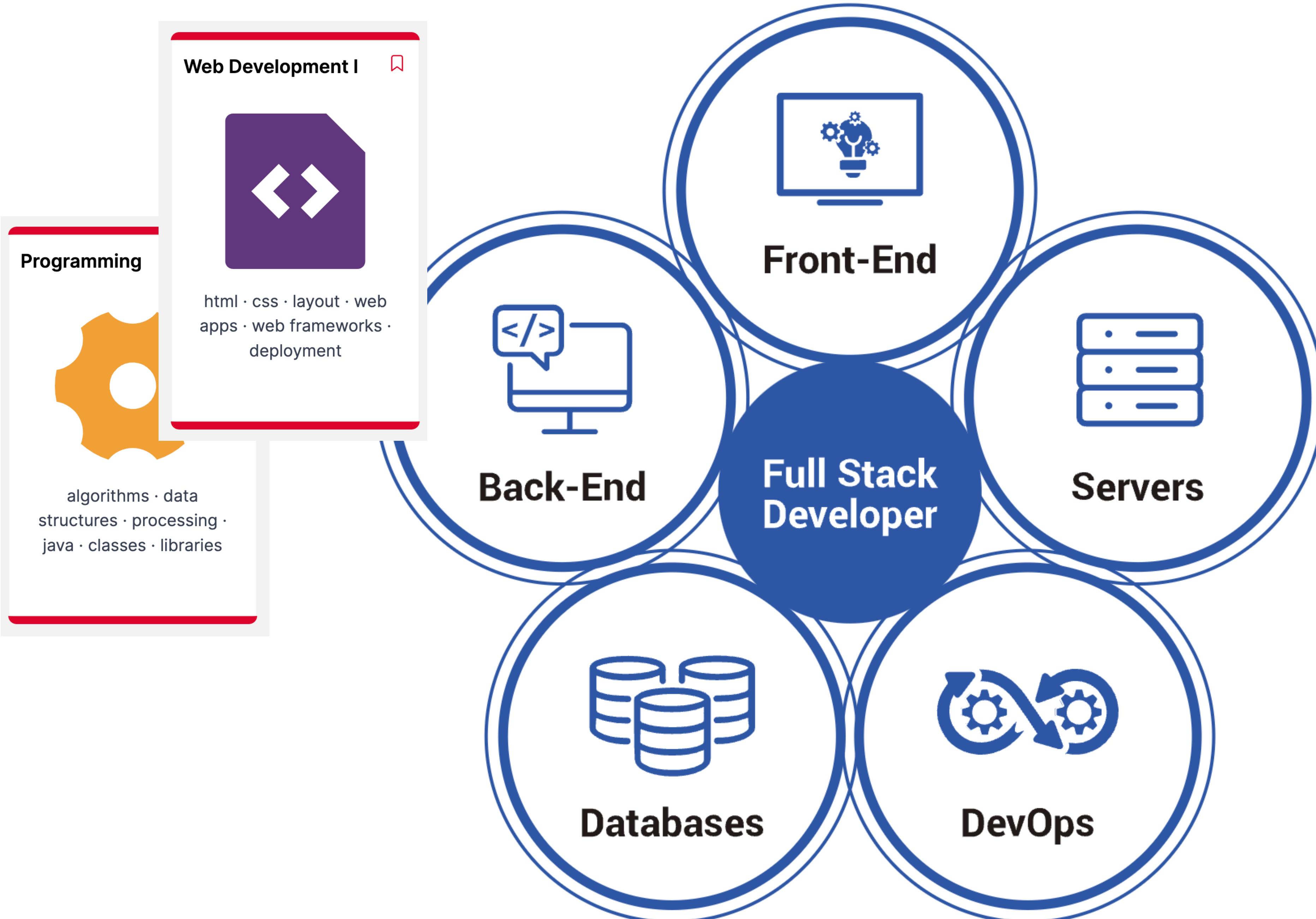
***“Outputs expected from the work placement would include a work placement report, a project ideally conducted in the work placement organisation...”***

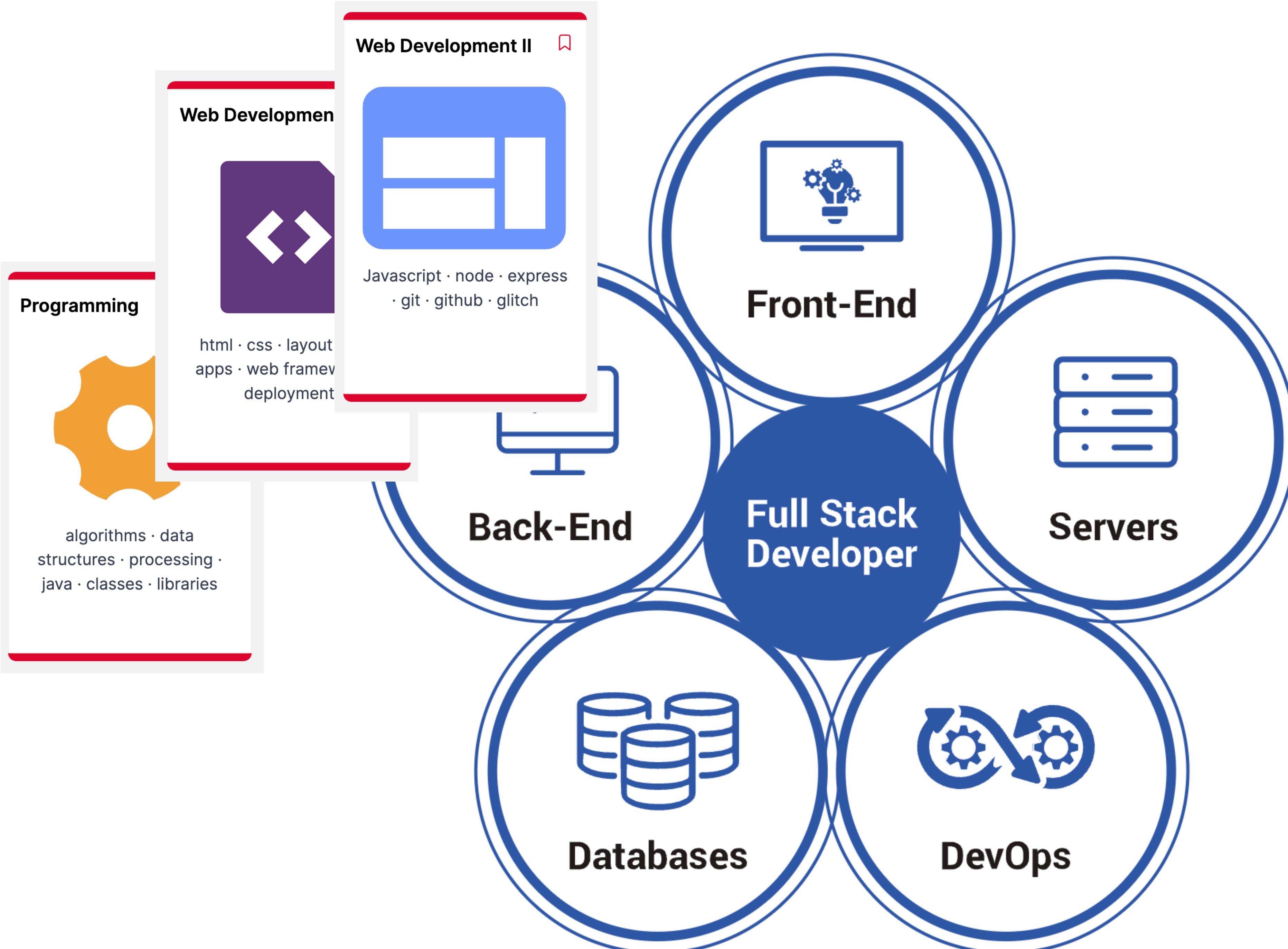
***“...academic and industry partners will cooperate in the provision of appropriate academic supervision resources for the duration of this work placement activity...”***

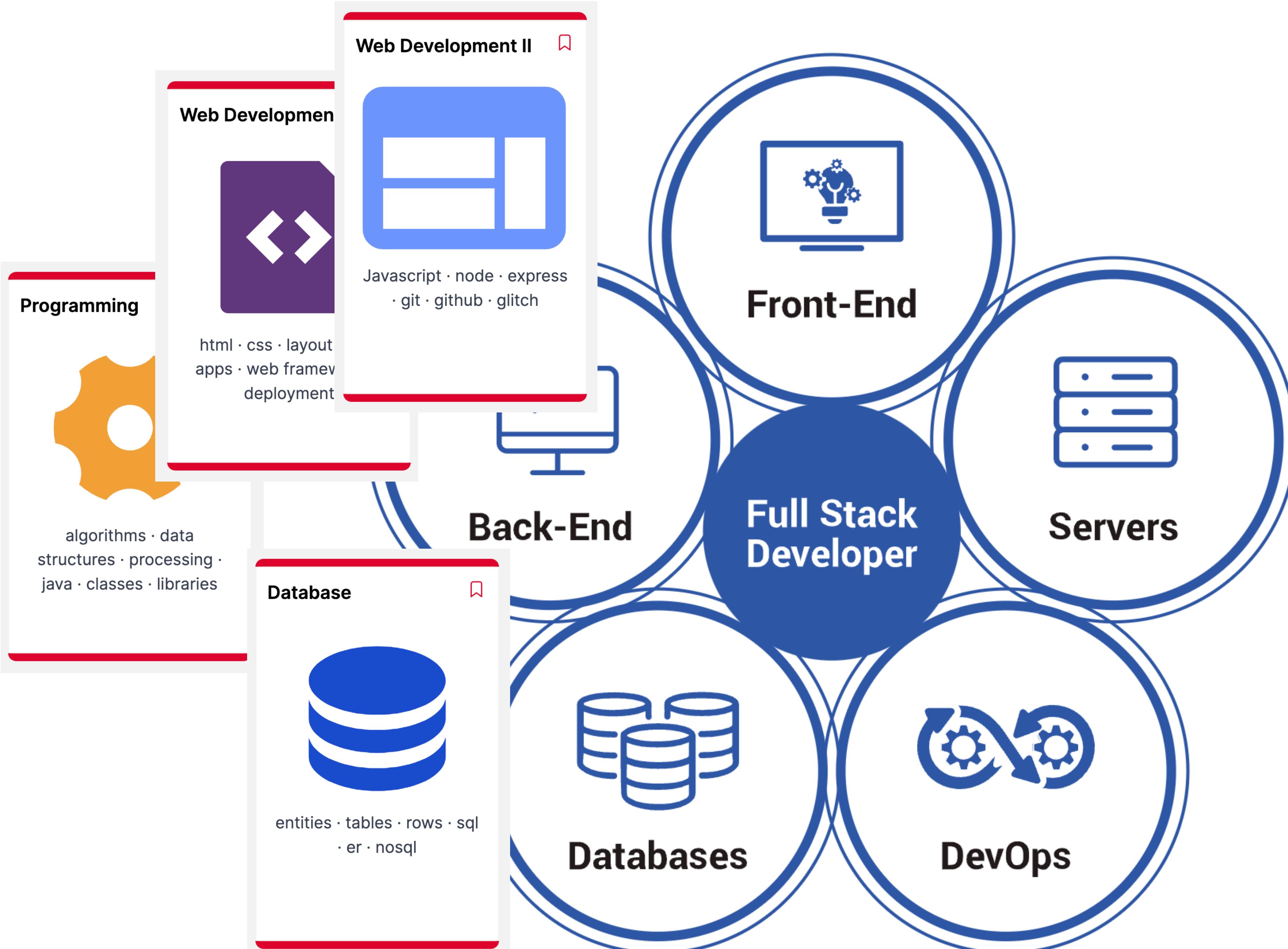
## 2. Programme Structure

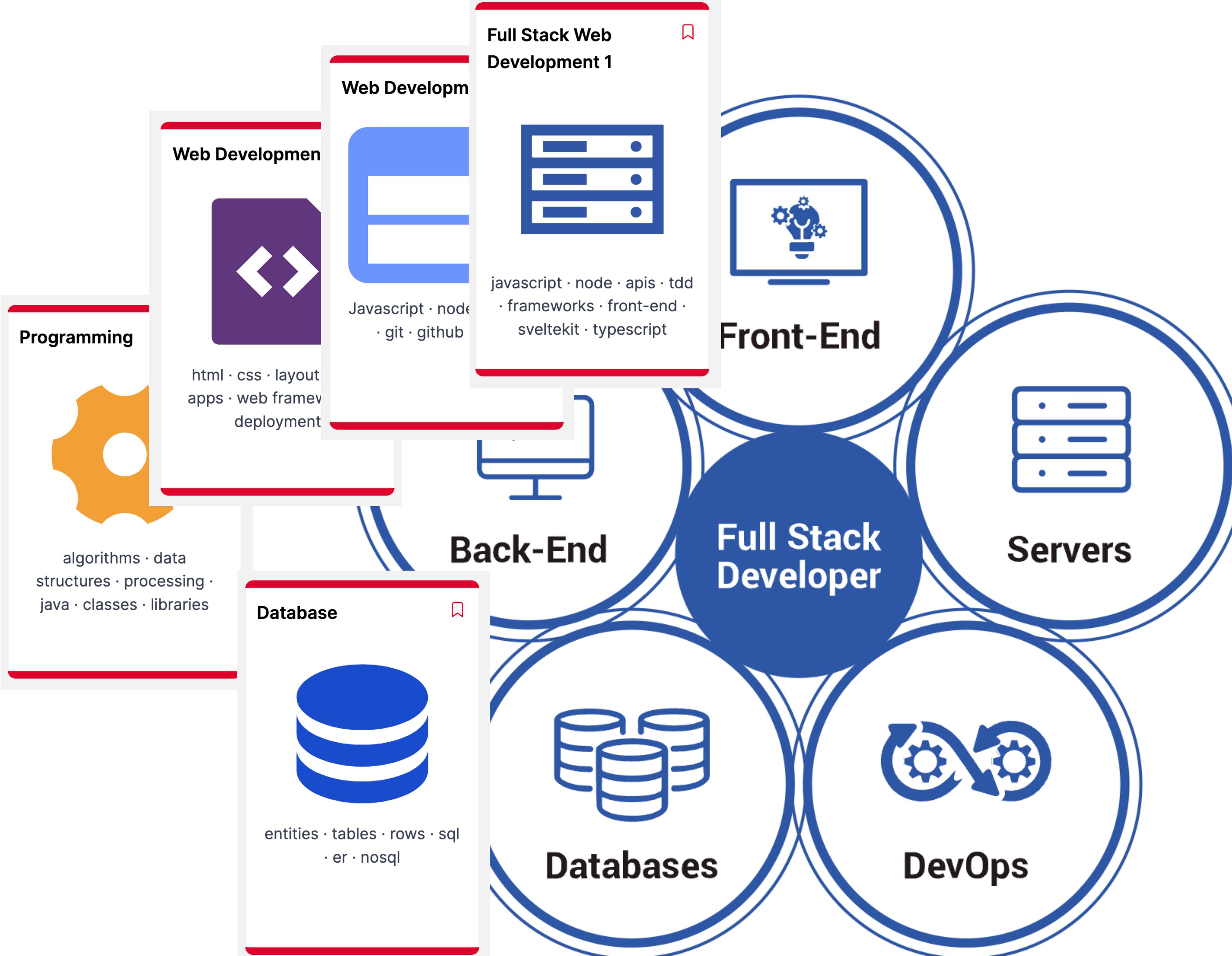


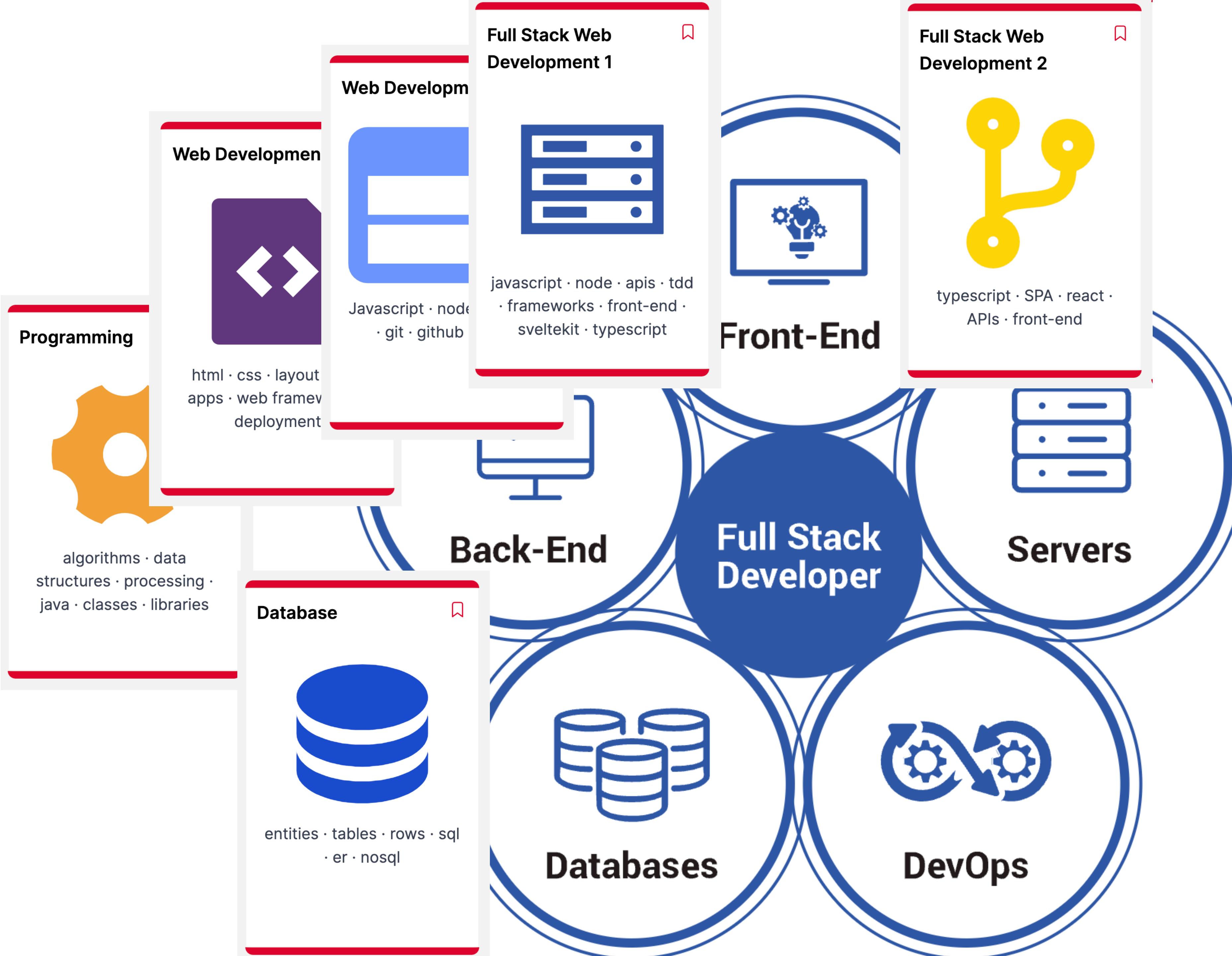


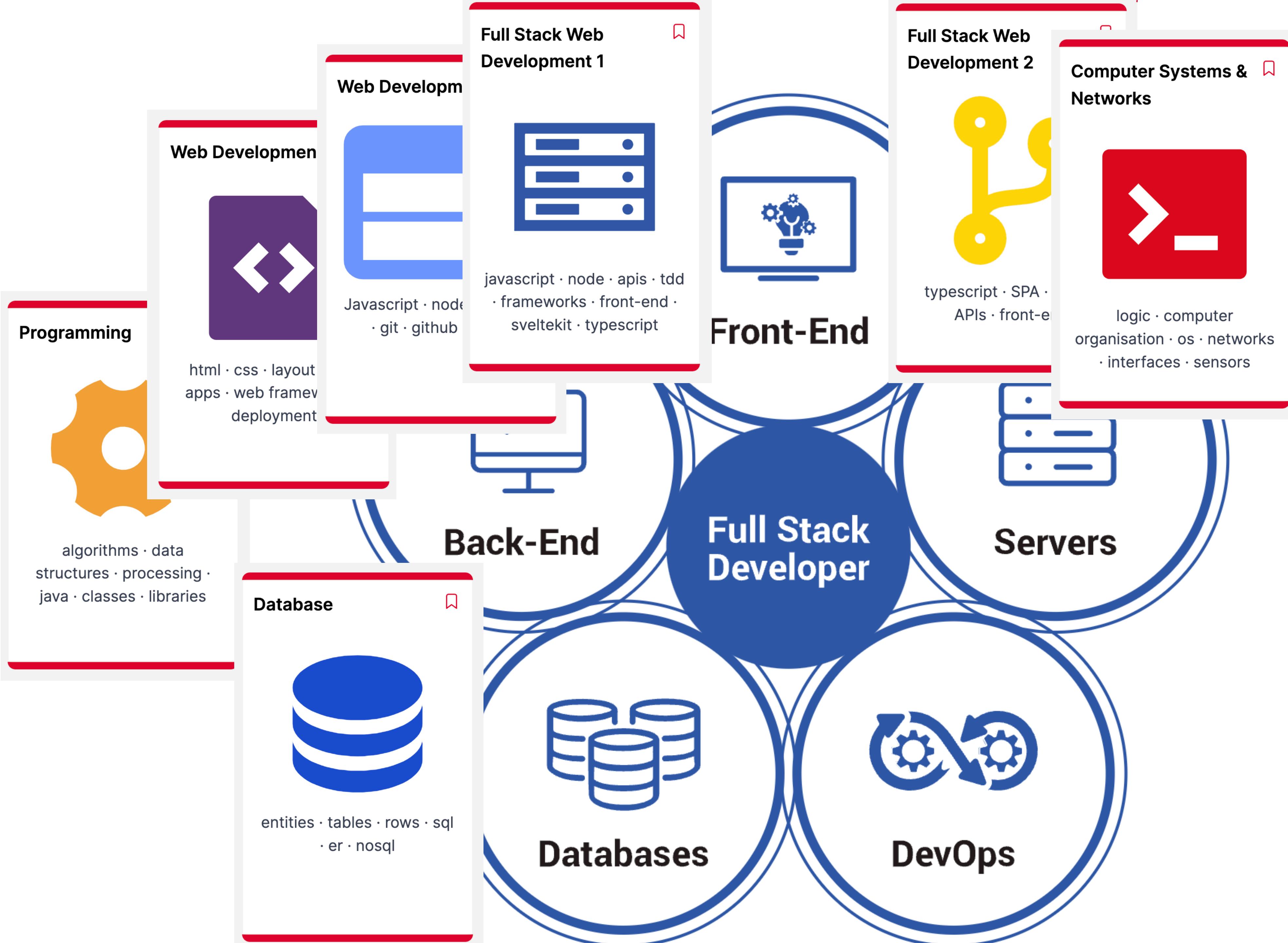


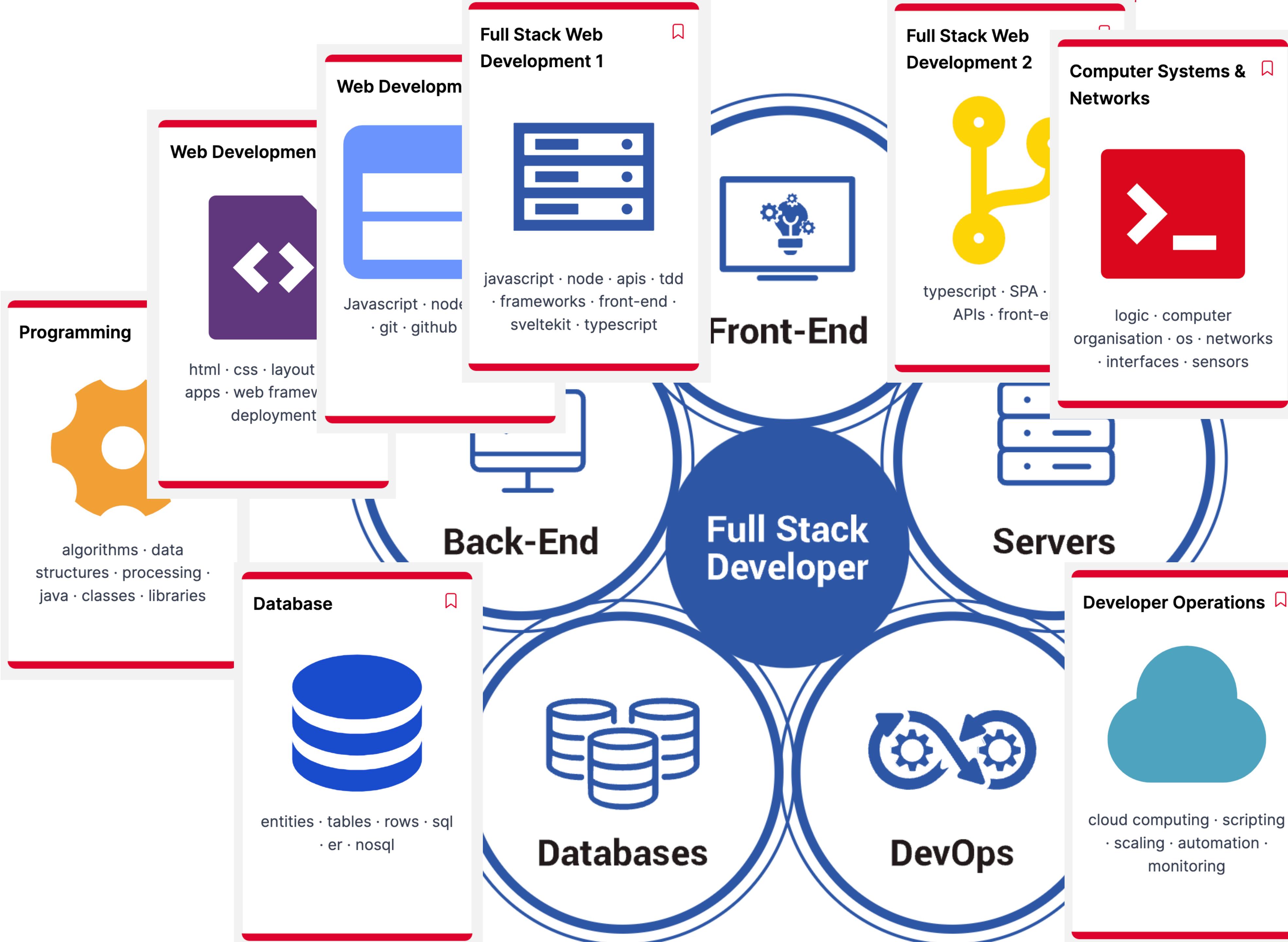


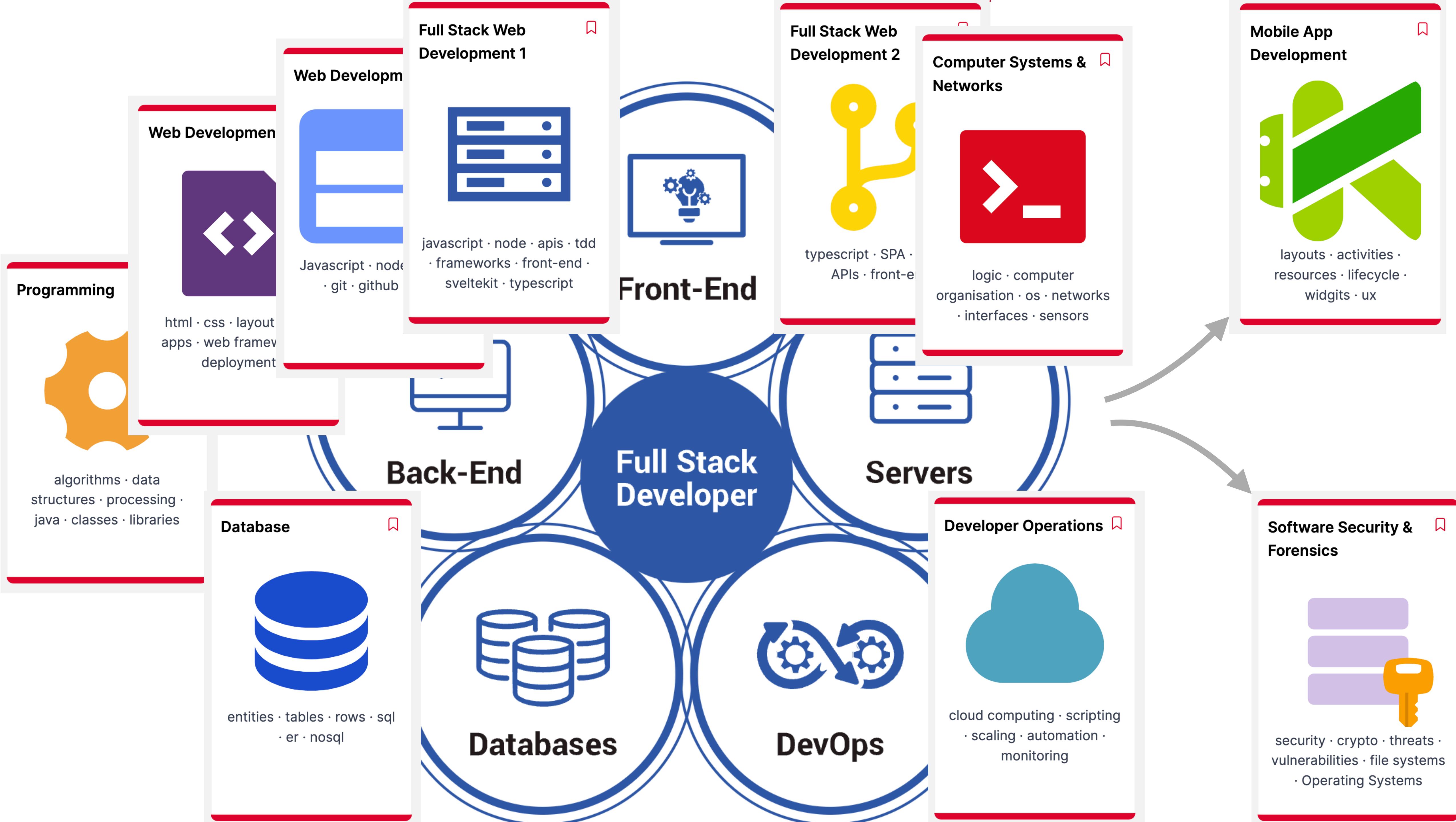


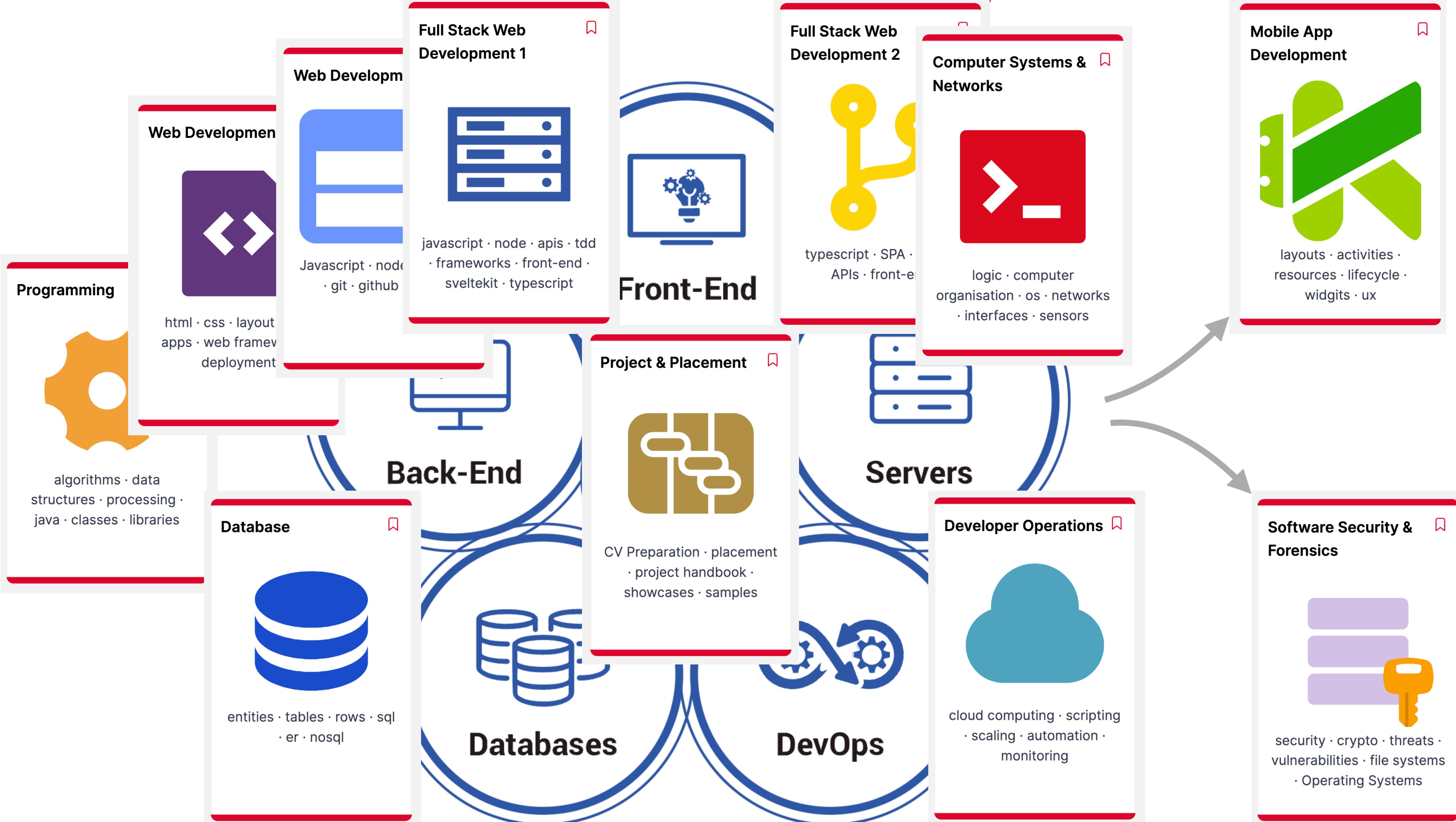


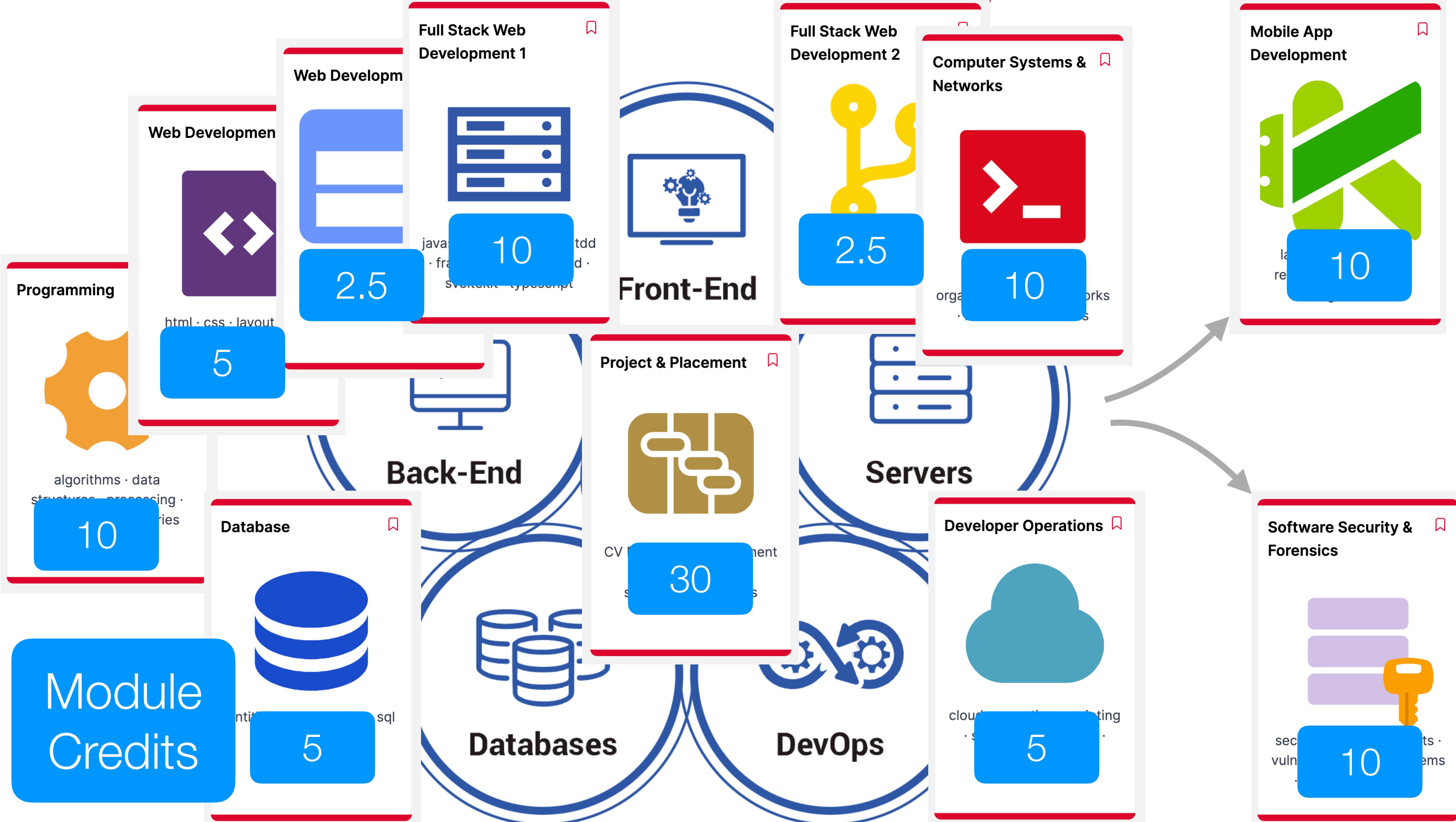




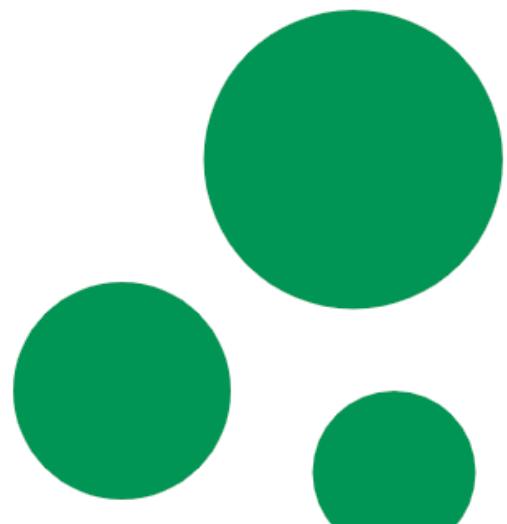






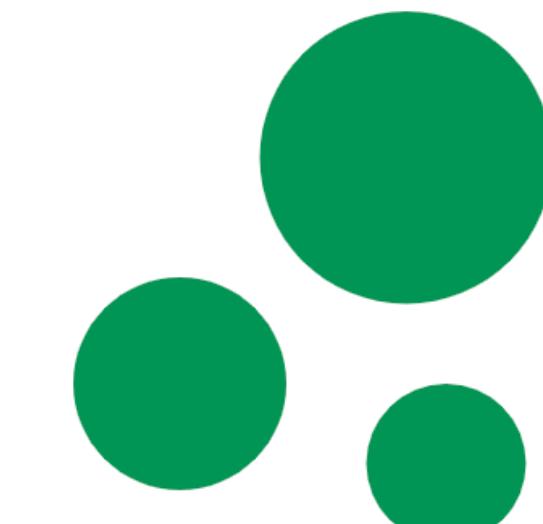


## Workshop One



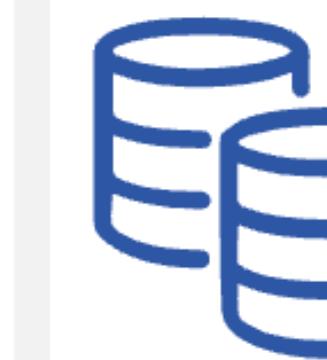
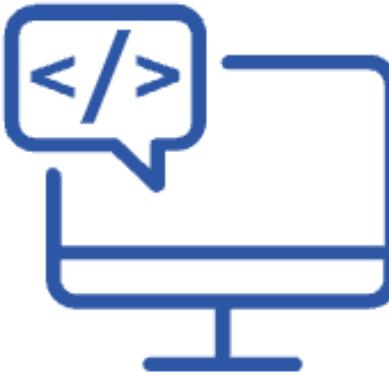
induction · structure · schedules · handbook

## Workshop Two



Introducing semester 2

## ·End

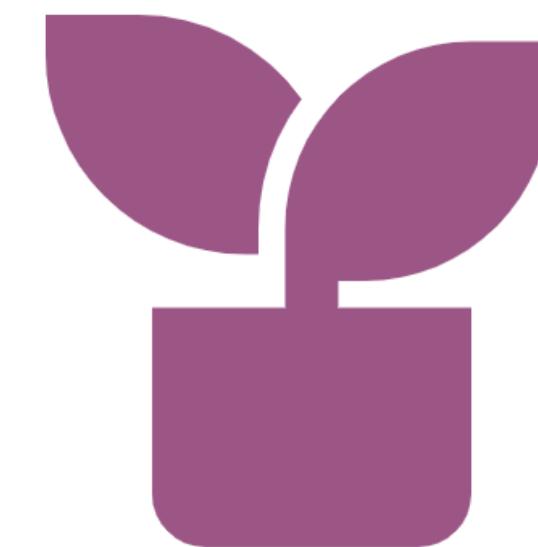


## Databases

## Front-End



## Life etc...



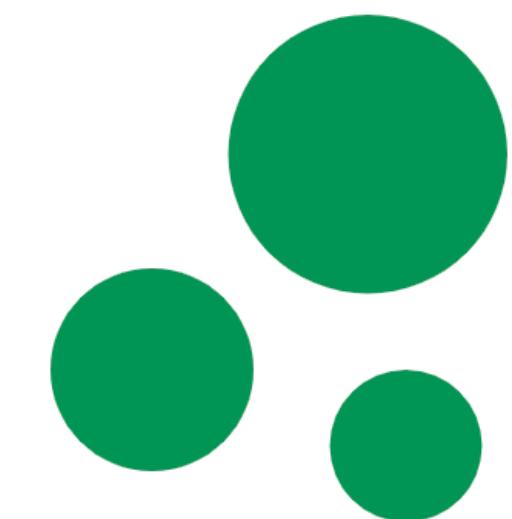
time · space · balance · experiences · tips · techniques

## Servers



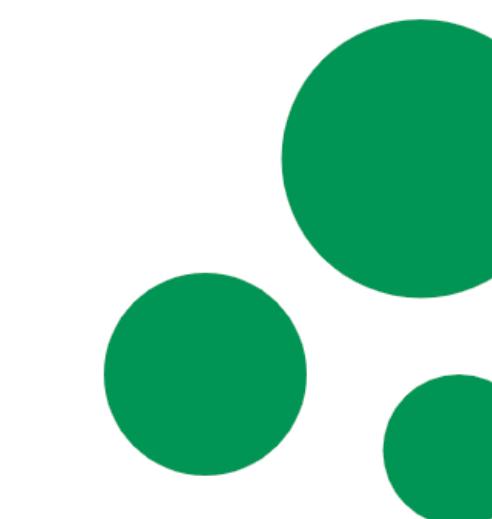
## DevOps

## Workshop Three



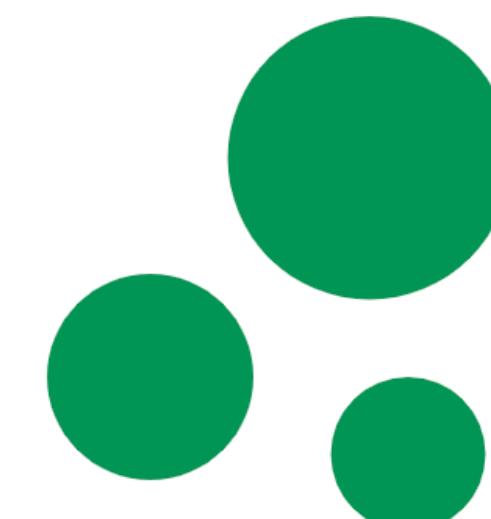
agile methods · cv preparation · introducing semester 3

## Workshop Four



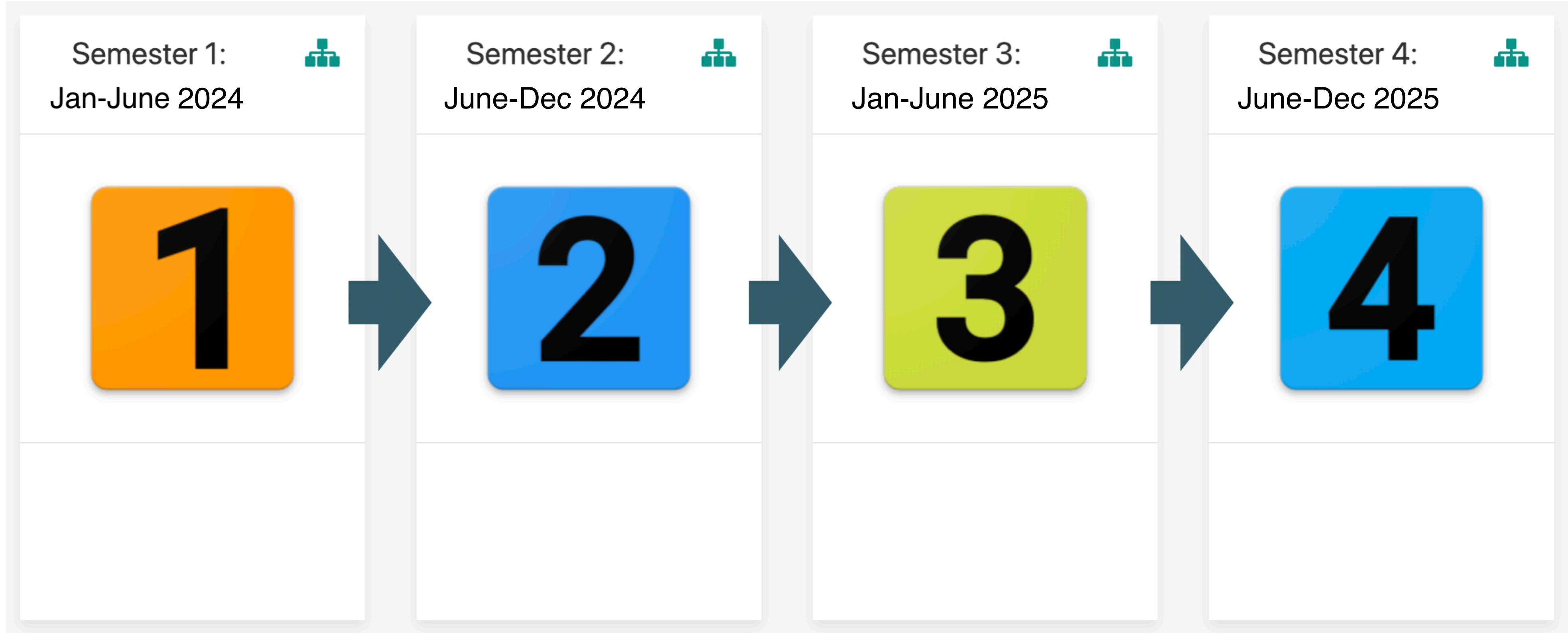
Introducing semester 4,  
Project, Placement

## Workshop Five



Friday January 12th, 2024

### 3. Semesters & Modules



June 2025 - May 2026

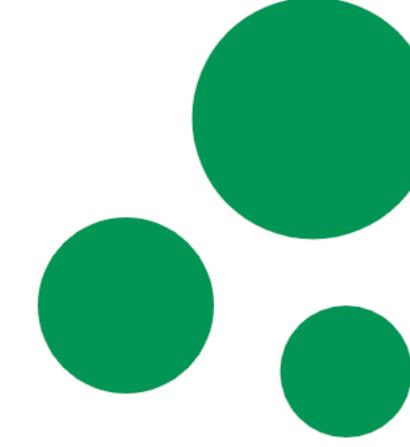


## Project/Work Placement

(Work Placement 4-6 months)

Semester 1: January - June 2024

**Workshop One** 



induction · structure ·  
schedules · handbook

**Web Development I** 



html · css · layout · web  
apps · web frameworks ·  
deployment

**Programming** 

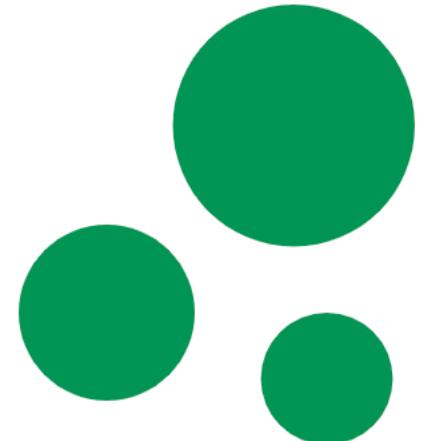


algorithms · data  
structures · processing ·  
java · classes · libraries

*“..a broad immersive set of modules in the fundamentals of computing covering **software development, systems analysis & testing, databases, architecture, OS & networking, web design / user-experience..”***

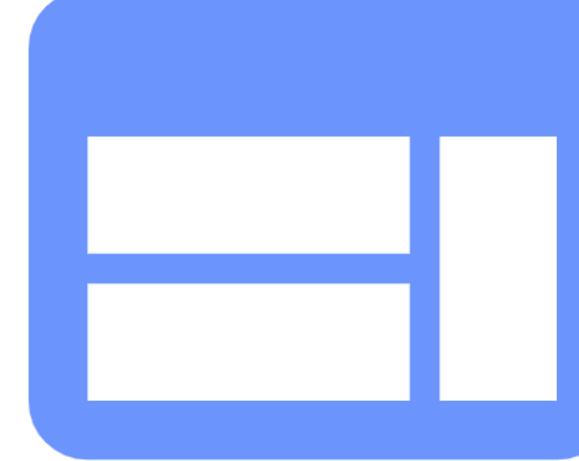
Semester 1: June - December 2024

**Workshop Two** 



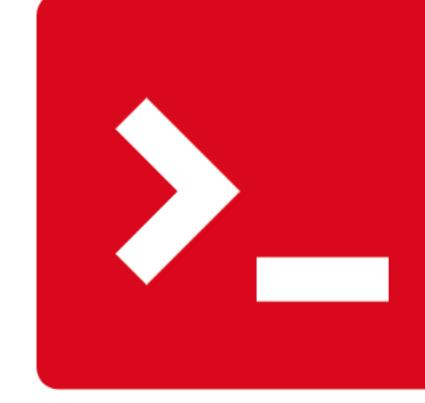
Introducing semester 2

**Web Development II** 



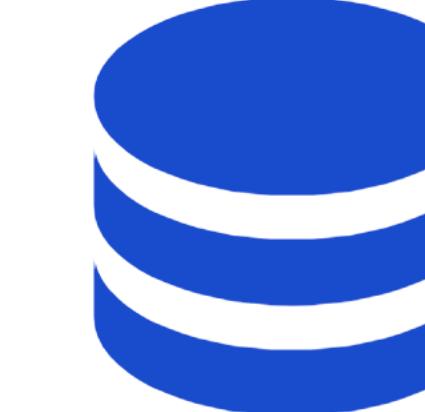
Javascript · node · express  
· git · github · glitch

**Computer Systems & Networks** 



logic · computer  
organisation · os · networks  
· interfaces · sensors

**Database** 

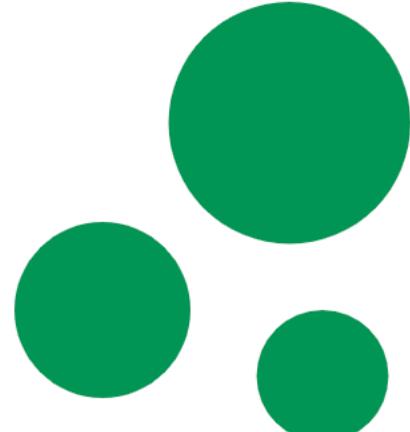


entities · tables · rows · sql  
· er · nosql

*“..a broad immersive set of modules in the **fundamentals of computing** covering software development, systems analysis & testing, **databases, architecture, OS & networking**, web design / user-experience..”*

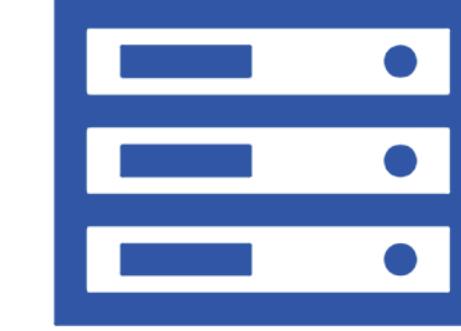
Semester 3: January - June 2025

**Workshop Three** 



agile methods · cv preparation · introducing semester 3

**Full Stack Web Development 1** 



javascript · node · apis · tdd · frameworks · front-end · sveltekit · typescript

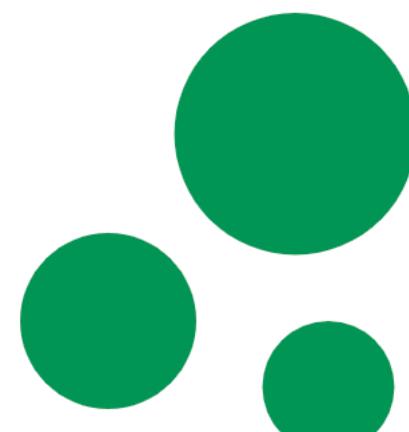
**Developer Operations** 



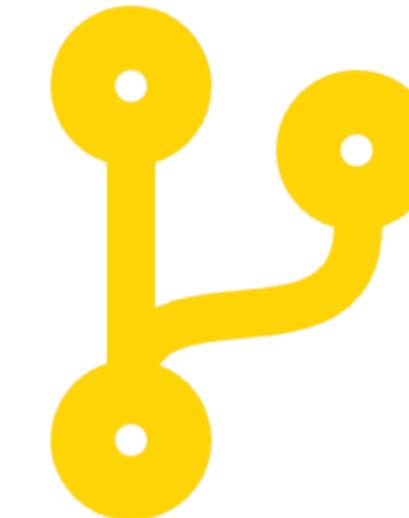
cloud computing · scripting · scaling · automation · monitoring

*“... students are expected to take a specialisation which reflects their own strengths as demonstrated on the programme to date...”*

Semester 4: June - December 2025

**Workshop Four** 

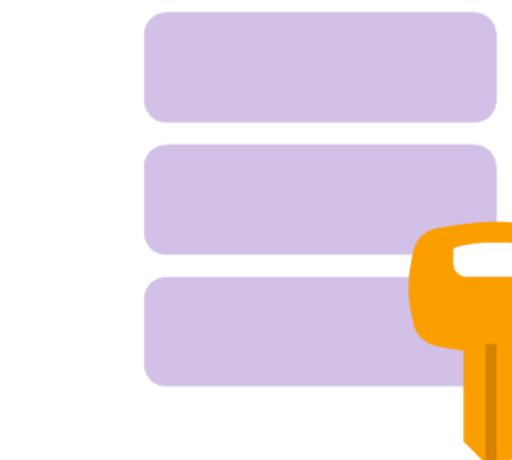
Introducing semester 4,  
Project, Placement

**Full Stack Web  
Development 2** 

typescript · SPA · react ·  
APIs · front-end

**Mobile App  
Development** 

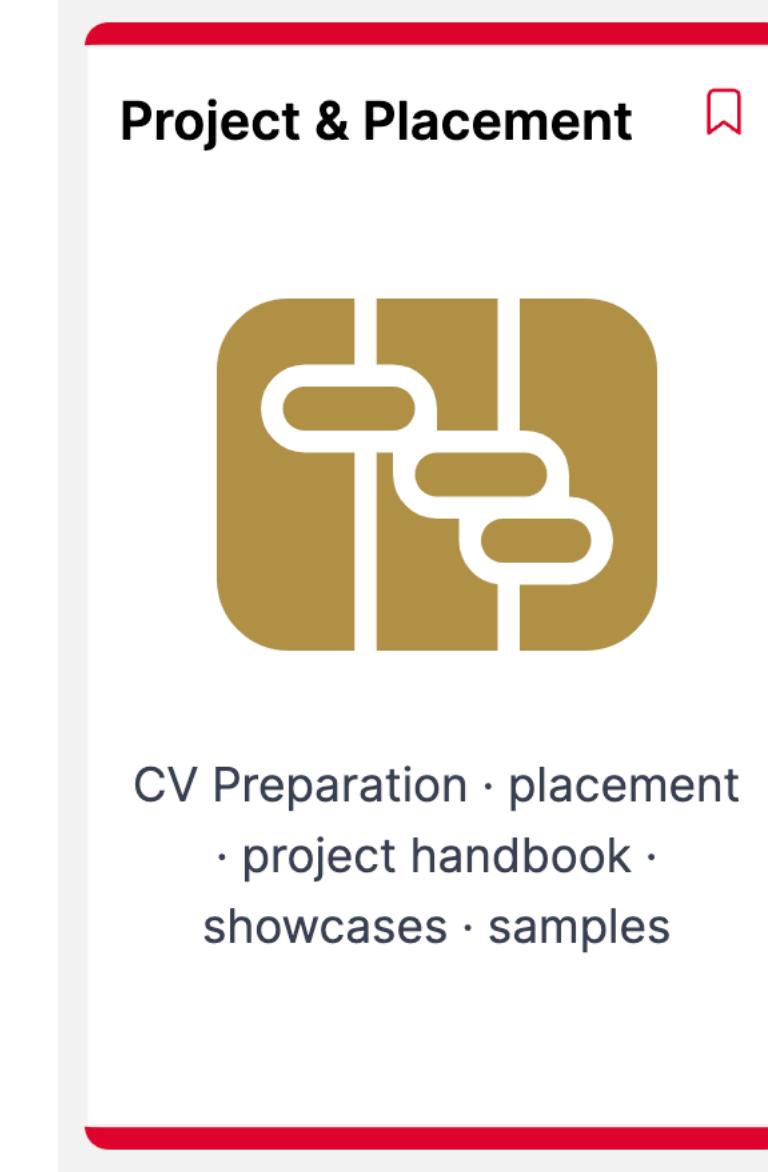
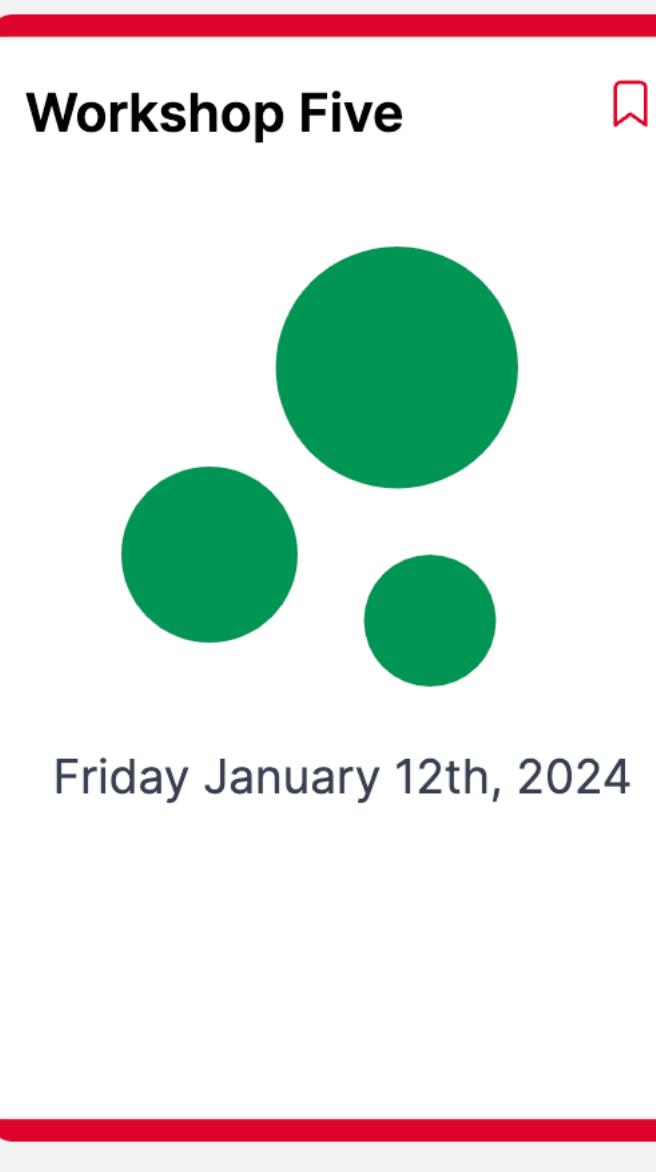
layouts · activities ·  
resources · lifecycle ·  
widgits · ux

**Software Security &  
Forensics** 

security · crypto · threats ·  
vulnerabilities · file systems  
· Operating Systems

*“... students are expected to take a specialisation which reflects their own strengths as demonstrated on the programme to date...”*

Semester 4: September 2025 - May 2026



*"Internships or work placements are seen as crucial to providing graduates with the context and confidence in their new knowledge..."*

## 4. Calendar, Timetable & Assessment Sequencing



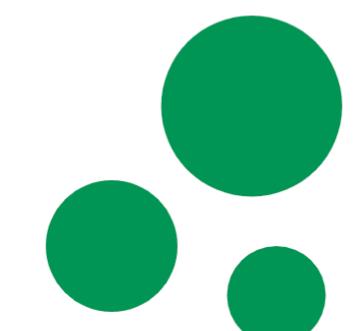
Semester 1  
2024

<b>Programming</b>	<b>Web Development I</b>
algorithms · data structures · processing · java · classes · libraries	html · css · layout · web apps · web frameworks · deployment

	S	M	T	W	T	F	S
Week	1	1	2	3	4	5	6
January	7	8	9	10	11	12	13
	14	15	16	17	18	19	20
	21	22	23	24	25	26	27
	28	29	30	31	1	2	3
February	reading	4	5	6	7	8	9
	11	12	13	14	15	16	17
	18	19	20	21	22	23	24
	reading	25	26	27	28	29	1
March	6	3	4	5	6	7	8
	10	11	12	13	14	15	16
	17	18	19	20	21	22	23
	Easter 1	24	25	26	27	28	29
April	Easter 2	31	1	2	3	4	5
	7	8	9	10	11	12	13
	14	15	16	17	18	19	20
	21	22	23	24	25	26	27
May	12	28	29	30	1	2	3
	5	6	7	8	9	10	11
	12	13	14	15	16	17	18
	19	20	21	22	23	24	25
June	13	26	27	28	29	30	31
	2	3	4	5	6	7	8
	9	10	11	12	13	14	15
	16	17	18	19	20	21	22
July	17	23	24	25	26	27	28
	30	1	2	3	4	5	6

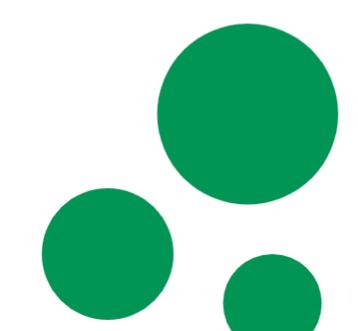
<b>Web Development II</b>
Javascript · node · express · git · github · glitch

Workshop One



induction · structure · schedules · handbook

Workshop Two

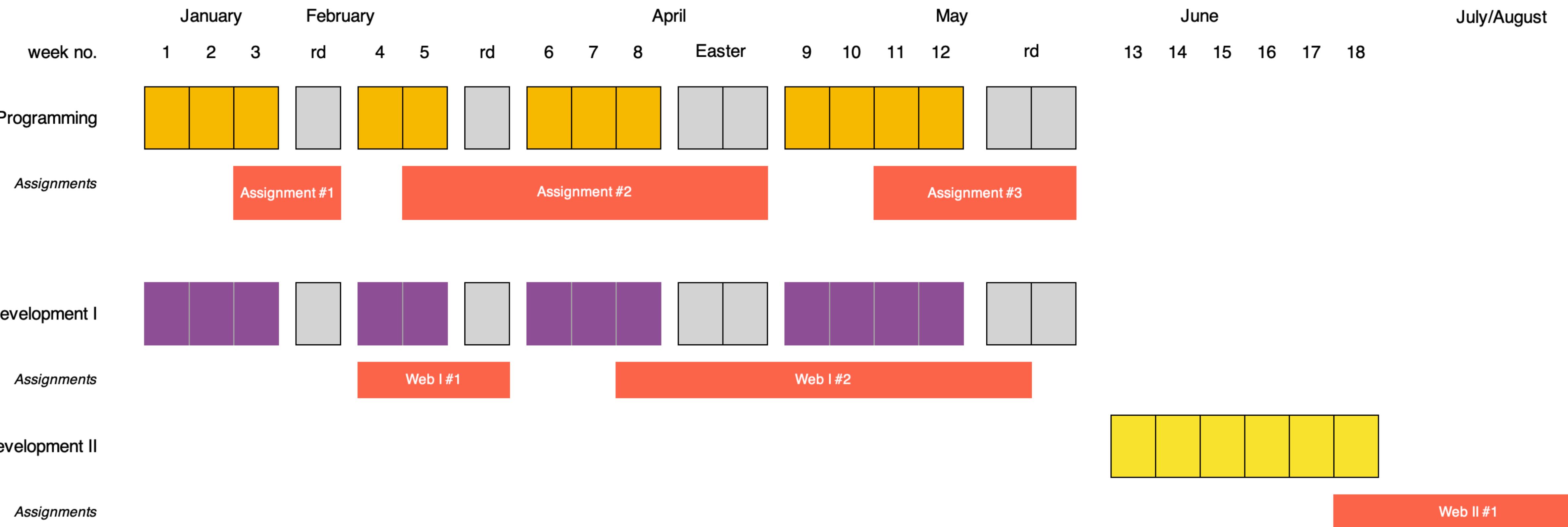


Introducing semester 2

# Weekly Webinar Schedule

MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
10:45				10:45
12:15  Programming <i>Webinar</i> 12:15-2:00		12:15  Programming <i>Webinar</i> 12:15-2:00		12:15  Web Development <i>Webinar</i> 12:15-2:00
2:00				13:45
15:15				15:15

# Semester 1 Assessment Schedule



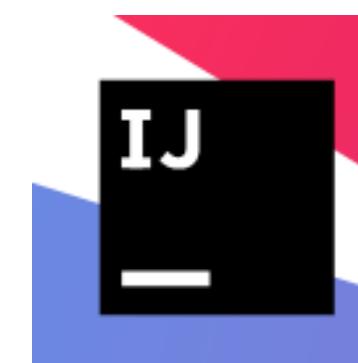
## 4. Module Summaries

# Programming



algorithms • data  
structures • processing •  
java • classes • libraries

- Apply core problem solving approaches suitable to the programming discipline to build algorithms.
- Construct small applications using standard sequence, conditional and iterative control structures. Change and expand small applications.
- Construct small applications that use simple UI, computation and data structures.
- Apply techniques to effectively test, debug and document small applications.
- Defend and explain how the above applications work.
- Apply problem-solving strategies to various computing problems of increasing complexity.
- Plan, code, test and document applications using advanced programming constructs and data structures



# Web Development 1

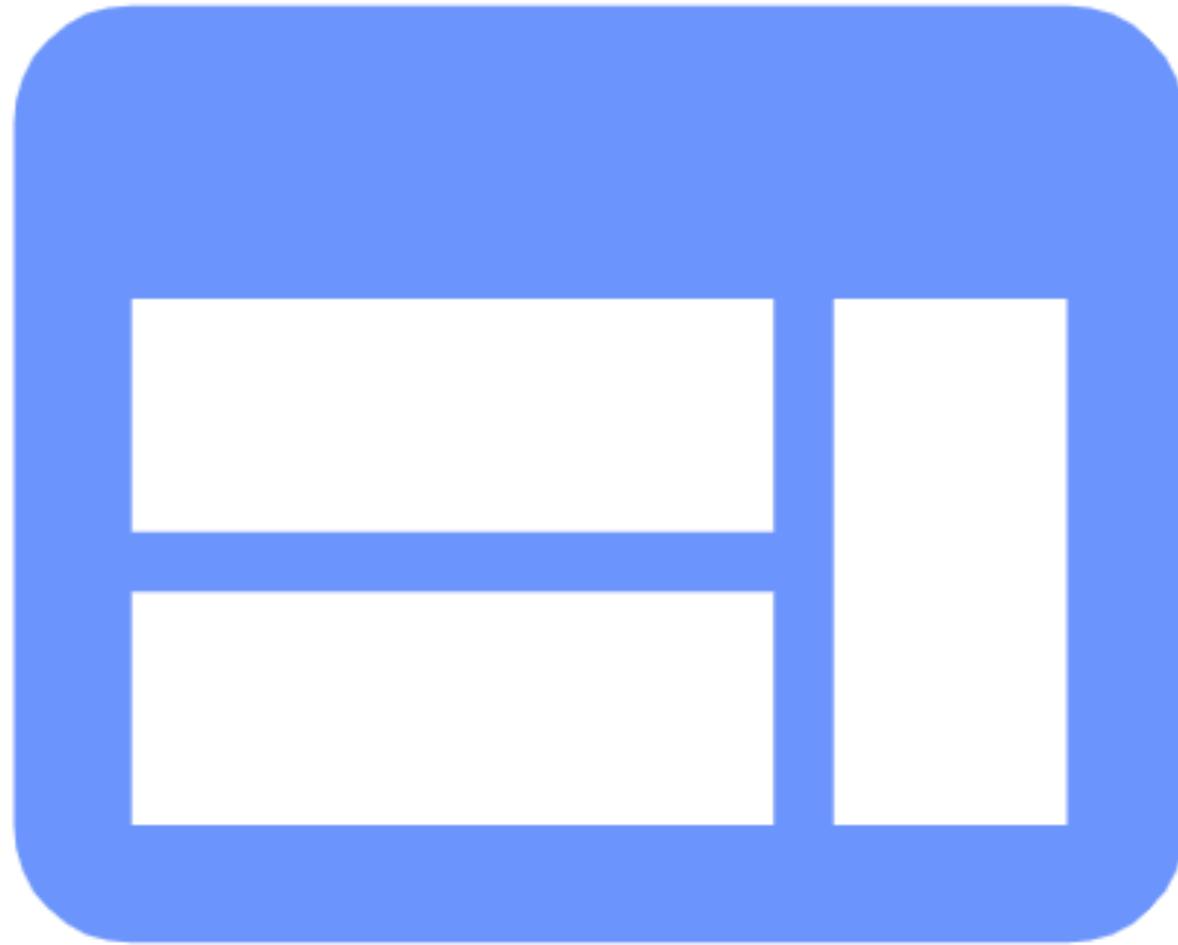


html · css · layout · web  
apps · web frameworks ·  
deployment

- Understand the fundamentals of the HTML markup language.
- Understand the role of Human Computer Interaction and manipulate CSS to present HTML content.
- Be able to integrate HTML, CSS and Java script to structure simple web sites.
- Understand how a dynamic web page is generated and be familiar with the role of html templating techniques
- Understand the difference between a web site and a web app. Be able to design and implement a simple web app.
- Implement a simple Model View Controller application pattern for a web app.

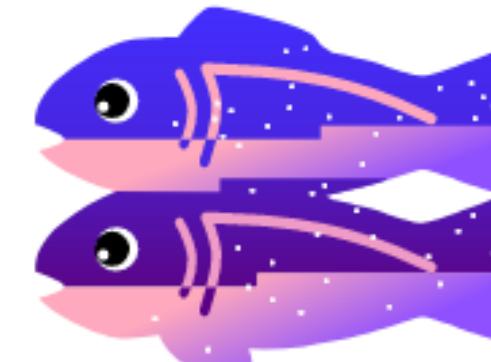


# Web Development 2



javascript · node · express  
· git · github · glitch

- Continue the journey into web application development
- Establish a competence in Javascript programming language
- Explore the basics of the Node.js framework
- Use a simple JSON persistent storage database
- Design, build and deploy a complete web application using these tools
- Understand the role of Agile methods in this context

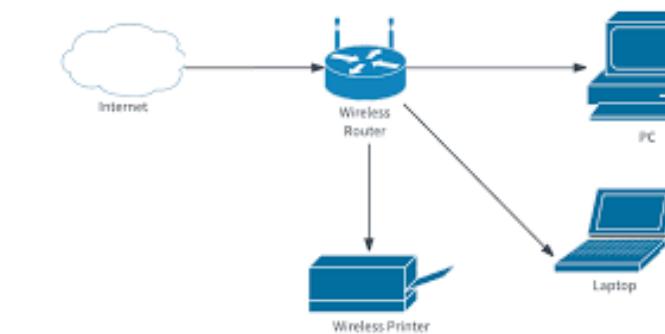


# Computer Systems & Networks



logic • computer  
organisation • os •  
networks • interfaces •...

- Identify and explain the role various hardware components play in a computer system.
  - Use an operating system on a chosen computer architecture.
  - Demonstrate an ability to configure systems using the command line.
  - Describe the memory management, process management and file management components of a modern operating system.
  - Explain basic concepts and theory of networked operating systems and virtualisation.
  - Configure a contemporary operating system (within a virtual machine environment)
  - Demonstrate competency in a limited set of utilities provided by a contemporary operating system.



# Databases



entities · tables · rows ·  
sql · er · nosql

MySQL™

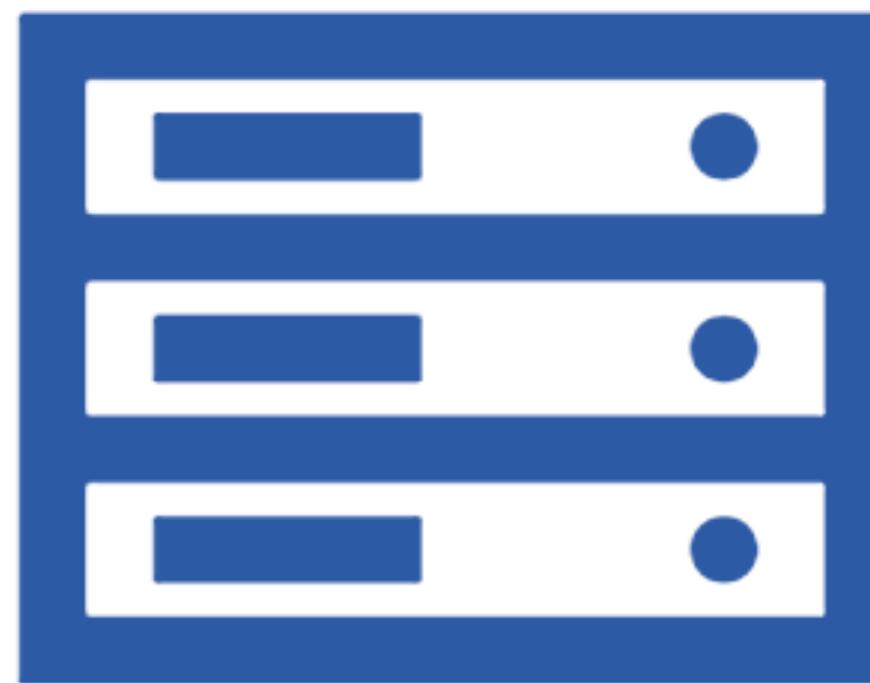
- Discuss the role of a database and its management system.
- Draw Entity Relationship (ER) diagram from an application problem and reproduce this diagram into a set of normalised relations, which are ready for database implementation.
- Design a NoSQL database suitable for a distributed environment with consideration of the CAP theorem.
- Gain an understanding of the physical database design process, its objectives and deliverables.
- Design and implement a database system



ORACLE®

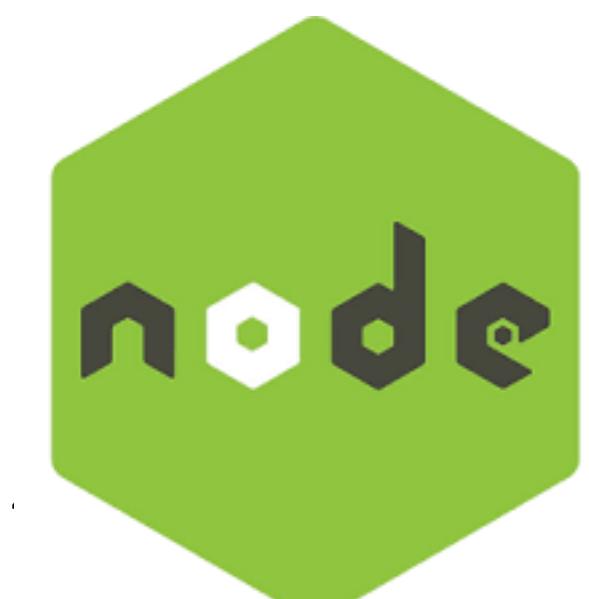
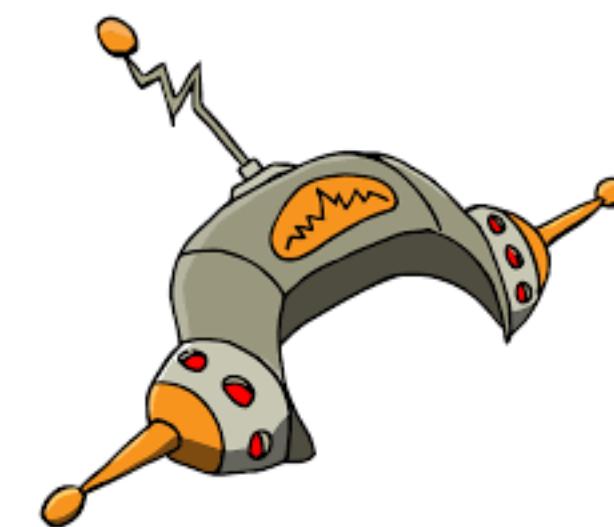
 mongoDB®

# Full Stack Development 1



javascript · node · apis ·  
tdd · frameworks · front-  
end · svelte

- Examine the key components of a server rendered web application and incorporate them into a running application.
- Use Model View Controller & related patterns in the implementation of a web project.
- Relate the request/response lifecycle, routing & session management in the context of a modern application framework.
- Model the user requirements and realize the model in a simple database.
- Apply best practice principles and patterns to the design and documentation of a web API.
- Apply best practice principles and patterns to the design of a medium-sized Single Page Web App.
- Develop an end-to-end web app that supports session management and persistence for a constrained functional requirement set.

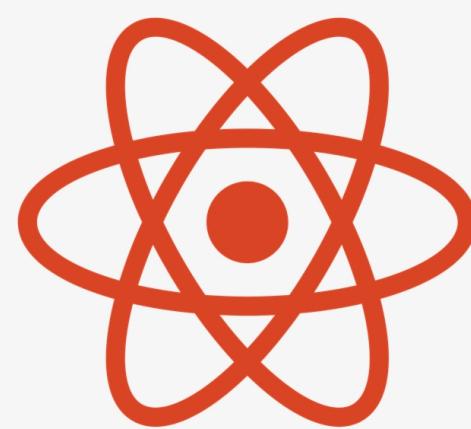


# Full Stack Development 2



SPA · react · APIs · front-  
end

- Introduce React + Storybook
- Explore the React component model
- Understand component navigation, lifecycle & routing
- Review the react methodology
- Select appropriate state management strategies & components

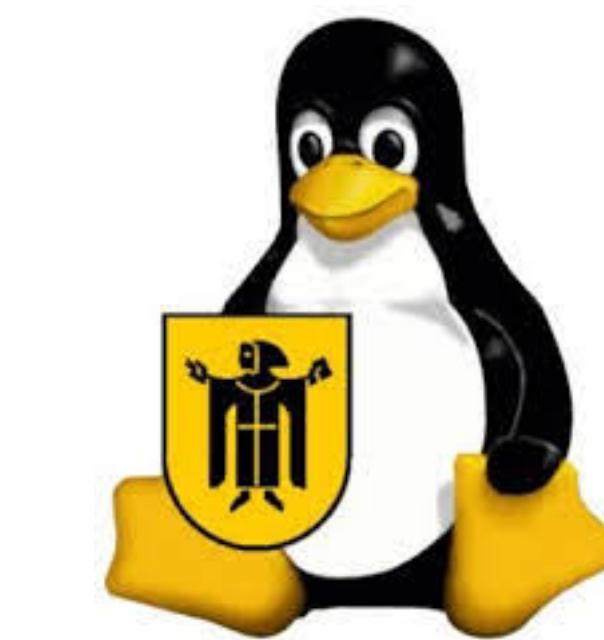


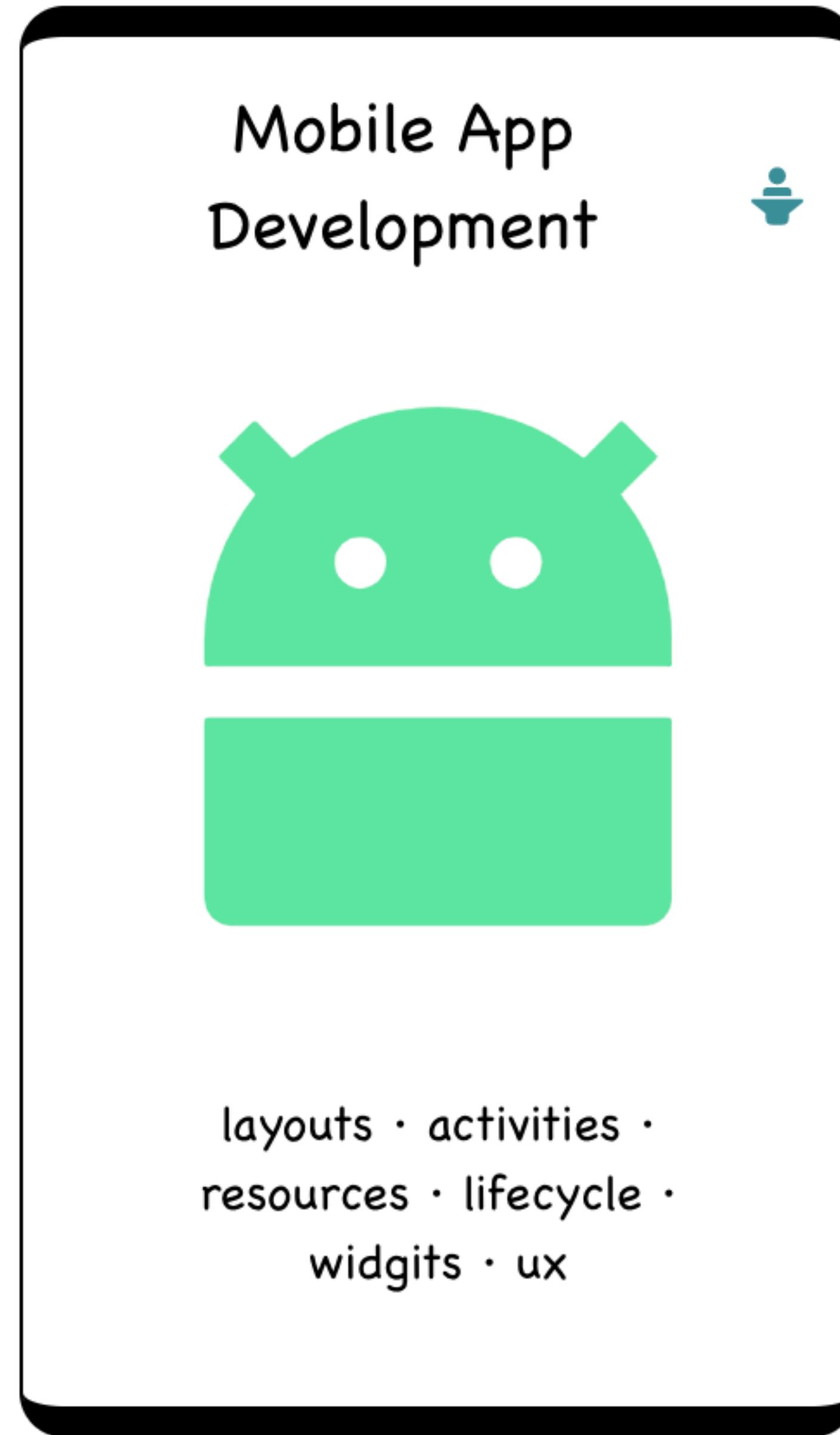
## Developer Operations



cloud computing · scripting  
· scaling · automation ·  
monitoring

- Build, configure and manage essential network infrastructure services.
- Build, configure and manage essential application services.
- Deploy a network monitoring solution.
- Develop scripts to assist in the management and automation of modern network services.
- Configure appropriate security mechanisms, including firewall rules, encrypted services, and authentication.

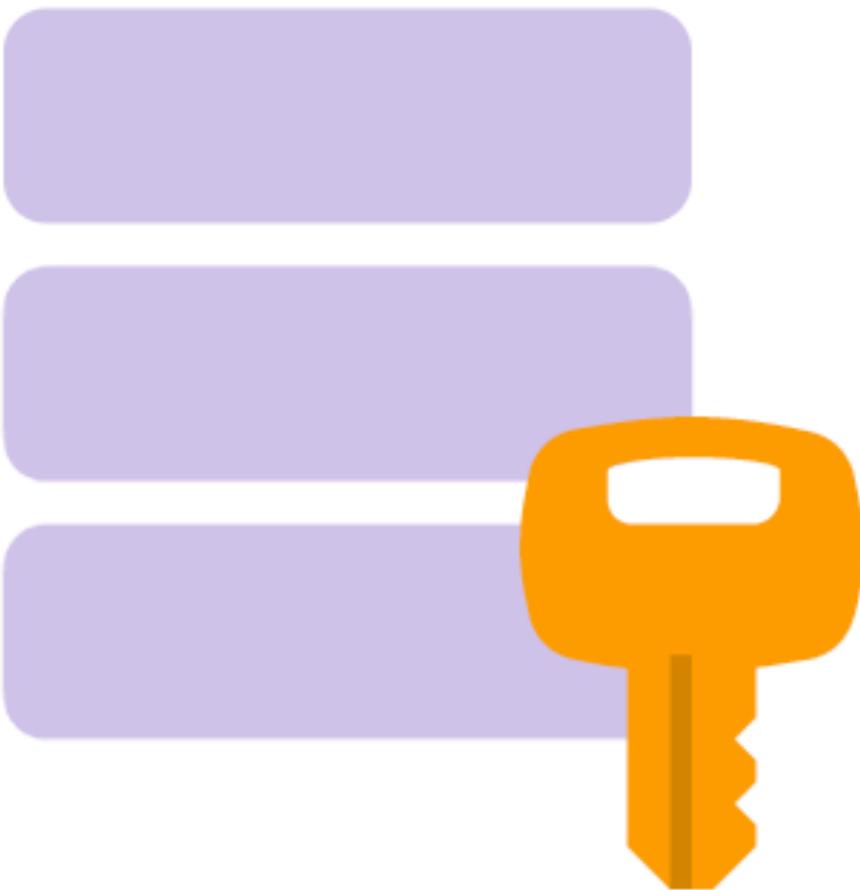




- Decompose an application into its constituent parts, including but not limited to: core application components, user experience resources, packaging.
- Design a coherent User Experience - using appropriate tools, practices and guidelines - for a moderately sized application. Produce a medium sized application, based on a limited set of design patterns.
- Manage the application lifecycle. Structure persistent storage on a device and reliably save and restore application state.
- Select the appropriate design patterns and tools in the development of complex mobile apps.
- Comment on the chosen mobile app framework and the underlying hardware components.
- Design and develop complex multi-screen mobile apps from concept through to completion using best practices and guidelines.
- Set up the interaction of an application with internal sensors and physical subsystems.
- Integrate a remote service API within an application, perhaps based on REST principles, to deliver aspects of its core features set.



# Software Security



security · crypto · threats  
· vulnerabilities



- Demonstrate specific security problems that can arise with web applications and how to address them.
- Compare and contrast alternative approaches to authentication in both enterprise and consumer-oriented web applications.
- Use a selection of best security practices in a web application.

# Opportunities for Further Study

- The development team are closely involved in the delivery of two potential follow-on graduate programmes:
  - MSc in Enterprise Software Systems
- These are mature courses, closely aligned with research at TSSG, with substantial enrolments in part-time mode from industry practitioners in the region.
- Successful candidates could continue their academic development in part-time or full-time capacity.



