This Industrial Project course will require the students to implement a medium-sized industrial project, contextualised for the work performed in the host company, focusing on the practical techniques of using a modern general-purpose programming language, such as C++, C# or Java.

The students will be using a version control system to store and manage their partial software product; they will use different system configurations to create different (stage-release) software products; and, where appropriate, they will use virtualisation technologies during software development and deployment.

LEARNING OUTCOMES – SUBJECT MASTERY

- In-depth understanding of the structured programming approach
- In-depth understanding of technologies and tools of software release management and version control
- In-depth understanding of a software version control tool to management software release, e.g. Git
- In-depth understanding of virtualisation technologies for software development, deployment, and scalability, e.g. Docker
- Understand software testing principles and can practice them independently in an industrial context
- To understand the time and effort involved in implementing an industrially-based project

LEARNING OUTCOMES – PERSONAL ABILITIES

- To be able to relate and/or apply learned knowledge to work place computing projects, when appropriate
- To be able to work with others in a medium-sized software project
- To be able to identify, define, and analyse alternative project scenarios
- Take significant responsibility for their work and for a range of resources
- To be able to communicate effectively with colleagues at work place and extract and organise requirements effectively

SYLLABUS

- Structured Programming Concepts
F28JP Industrial Project: Structured Prog

- System configuration and Release Management: System configuration process; Version control (e.g. Git); Software release management methods, (e.g. itSMF, ITIL); Software release management tools (e.g. Puppet, Plutora); Virtual Machine (e.g. Docker, VirtualBox) usage for development, deployment, and scalability on distributed and Cloud systems
- Software testing: Systematic testing; Test-driven design
- Industrial project development (incl. project scope and design, planning, requirements engineering, system implementation, software testing and evaluation, and critical assessment)

COURSE RELATIONSHIPS
N/A

LOCATION AND ASSESSMENT METHODS

<table>
<thead>
<tr>
<th>Edi</th>
<th>SBC</th>
<th>Ork</th>
<th>Dub</th>
<th>Malay</th>
<th>IDL</th>
<th>COLL</th>
<th>ALP</th>
<th>OTH</th>
<th>Method</th>
<th>Weight</th>
<th>Exam Mins</th>
<th>Type</th>
<th>Diet</th>
<th>Synoptic Course</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Coursework</td>
<td>100</td>
<td>Assessment</td>
<td>Semester 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Coursework</td>
<td>100</td>
<td>Reassessment</td>
<td>Semester 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>