Course descriptor F21RS

Course code	**F21RS**
Course title	Rigorous Methods for Software Engineering
Credits	15
School	Mathematics and Computer Sciences
SCQF Level	11
Semester	1
Aims	To provide knowledge and understanding of tools and techniques which support rigorous software engineering
Syllabus	The course addresses the challenges of engineering safe and secure software systems. It covers a range of rigorous processes and formal methods that support the development of high integrity software systems. From modelling and reasoning about designs through to code, students will experience a range of state-of-the-art static analysis tools and techniques. While theory based, the course has a strong practical element, drawing upon industrial case study material where appropriate.

Learning Outcomes	
Subject Mastery	• A detailed and integrated knowledge and understanding of a range of rigorous processes and formal methods that support the development of high integrity software systems. • Critical understanding of the relationship between code level annotations and high-level formal specifications. • Extensive knowledge of the mechanisms that underlie advanced static analysis techniques. • To be able to demonstrate a critical understanding of the relationship between code level annotations and flow analysis techniques. • To be able to demonstrate a critical understanding of program proof and how it can be used to provide strong formal correctness guarantees.
Personal Abilities	• Conceptualize and define new abstract problems within the context of automated software development. • Deal with complex issues and make informed judgements in situations in the absence of complete or consistent data. • Exercise substantial autonomy, initiative and creativity in the application of software engineering techniques. • Demonstrate critical reflection. (PDP) • Communicate with peers, more senior colleagues and specialists. (PDP)

Assessment method	60% written examination, 40% coursework