

## 6SENG006W Concurrent Programming Coursework (2025/26)

<b>Module leader</b>	Guhanathan Poravi ( <a href="mailto:guhanathan.p@iit.ac.lk">guhanathan.p@iit.ac.lk</a> )
<b>Unit</b>	Coursework Deferral / Referral
<b>Weighting:</b>	50%
<b>Qualifying mark</b>	30%
<b>Description</b>	To develop:  A concurrent (multi-threaded) Java program to implement based off 2 given scenarios.
<b>Learning Outcomes Covered in this Assignment:</b>	The coursework assesses learning outcomes: LO1, LO2, LO3 & LO4.
<b>Handed Out:</b>	N/A
<b>Due Date</b>	<b>Before 13:00, 7 January, 2026</b>

<b>Expected deliverables</b>	<p>Electronic files:</p> <p>(a) The Java source code.</p> <p>(b) Vodcast – demonstrating the programs, explaining, and justifying your chosen solution.</p> <p>Source code files should be compressed into a single ZIP archive. The ZIP archive should be named using your surname &amp; "CW", e.g. "howells_CW.zip" Vodcast submitted separately on Blackboard.</p>
<b>Method of</b>	Online via Blackboard

<b>Submission:</b>	
<b>Type of Feedback and Due Date:</b>	<p>Verbal feedback in tutorials as the assessment progresses.</p> <p>Electronic via module's Blackboard bulletin board.</p> <p>Written feedback and marks 15 working days (3 weeks) after the submission deadline.</p> <p><b>All marks will remain provisional until formally agreed by an Assessment Board.</b></p>

### **Assessment regulations**

Refer to the "How you study" guide for postgraduate students for a clarification of how you are assessed, penalties and late submissions, what constitutes plagiarism etc.

### **Penalty for Late Submission**

If you submit your coursework late but within 24 hours or one working day after the specified deadline, 10 marks will be deducted from the final mark, as a penalty for late submission, except for work which obtains a mark in the range 40 – 49%, in which case the mark will be capped at the pass mark (40%). If you submit your coursework more than 24 hours or more than one working day after the specified deadline you will be given a mark of zero for the work in question unless a claim of Mitigating Circumstances has been submitted and accepted as valid.

It is recognised that on occasion, illness or a personal crisis can mean that you fail to submit a piece of work on time. In such cases you must inform the FST Registry in writing on a Mitigating Circumstances

(MC) form, giving the reason for your late or non-submission. You must provide relevant documentary evidence with the form. This information will be reported to the relevant Assessment Board that will decide whether the mark of zero shall stand. For more detailed information regarding University Assessment Regulations, please refer to the following website:  
<http://www.westminster.ac.uk/study/current-students/resources/academic-regulations>

#### Coursework Description

#### Scenario: 30 Marks

Eastminster University has experienced remarkable growth over the past three years, seeing considerable growth with the introduction of online and distance learning 5 years ago. What began as a small institution with around 5000 students has now expanded to more than 35000, spread across multiple campuses and online programmes. This rapid growth is projected to continue exponentially with anticipation of upwards of 100000 students in 2030. The University takes pride in its commitment to technological innovation and delivers all assignments through an online submission portal.

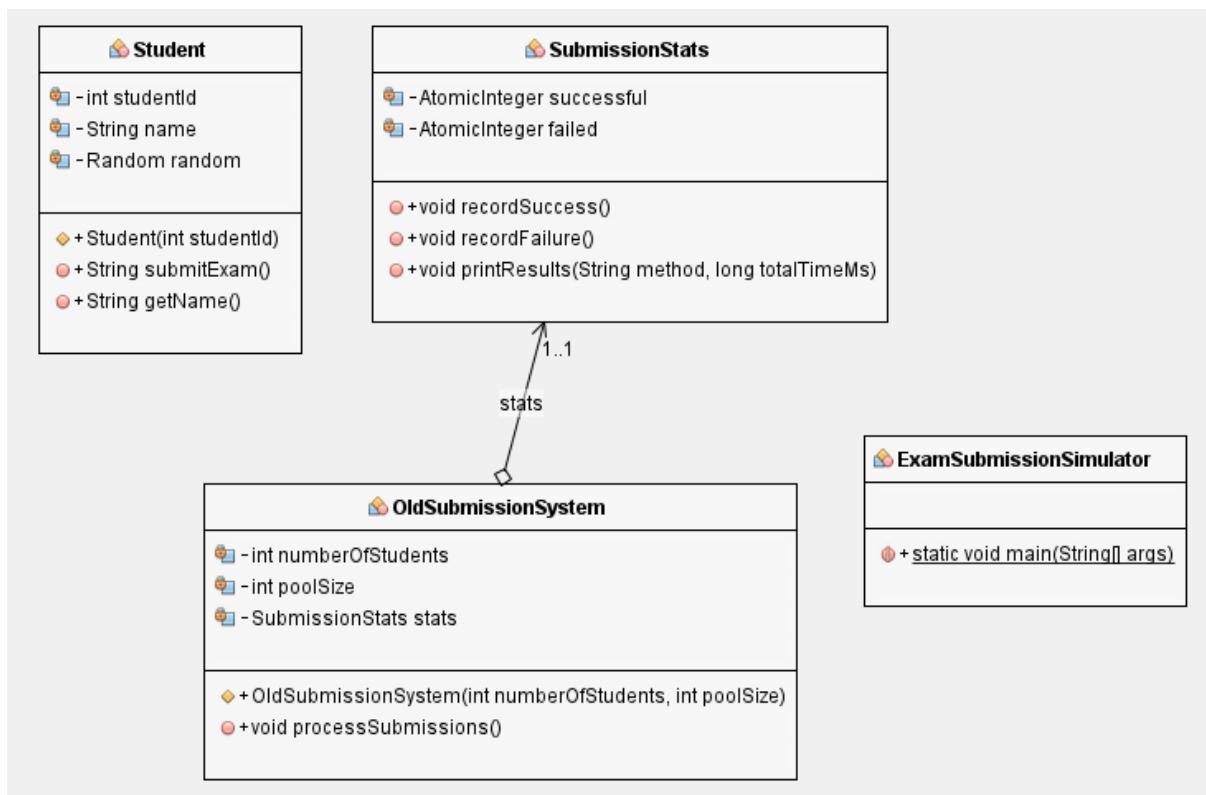
#### The Problem

The existing examination submission system was developed five years ago, when the University was considerably smaller. During the most recent examination period, the system proved entirely inadequate:

- Students reported waiting 20–30 minutes to submit their exams.
- Numerous submissions failed due to timeouts, causing significant anxiety just before deadlines.
- The IT Helpdesk was inundated with hundreds of support requests.
- Some students missed their submission deadlines through no fault of their own, as the system was unable to process files quickly enough.
- The Students' Union issued formal complaints on behalf of the affected students.

As a result, the Dean of Technology has received an urgent directive from the University Board:

"Resolve the issues with the submission system before the next semester, or we risk damaging our accreditation and losing the trust of our students."



Design & develop a solution that can handle this large number of submissions in preparation for the anticipated growth in such a way that it can scale.

Mark Scheme

### Task 1: Implement SubmissionStats Class (10 marks)

Create a thread-safe statistics tracking class following the class diagram.

#### Requirements:

- Track successful and failed submissions
- Use appropriate thread-safe data structures
- Implement a method to display results including:
  - Total time taken
  - Total students processed
  - Number of successful submissions
  - Number of failed submissions
  - Success rate percentage

### Task 2: Implement NewSubmissionSystem Class (20 marks)

Implement the main system that processes all student submissions concurrently.

#### Requirements:

- Handle 5000 and up to 100000 + students submitting simultaneously
- Handle exceptions gracefully
- Wait for all submissions to complete before finishing
- Measure and report total execution time
- Display individual submission results (success/failure messages)

### **Scenario 2 30 Marks**

Dr. Rebecca Patel, the newly appointed Chief Clinical Information Officer at Royal Manchester Hospital, sat in the emergency governance meeting reviewing last week's incident reports. The numbers painted a troubling picture:

*"We had 47 patients leave A&E without being seen last Tuesday. Average wait time hit 4 hours—well beyond our target of 95% seen within 4 hours. The system crashed twice during the evening shift. We're breaching targets, and worse, we're failing our patients."*

The problem was clear: Royal Manchester's current patient management system couldn't handle peak loads. Built on legacy technology from the early 2000s, it processed patients sequentially—one at a time. During busy periods, patients waited hours in A&E whilst the single-threaded system crawled through the queue. Consultants sat idle when no patients matched their speciality, whilst patients needing other specialists waited endlessly.

*"The Care Quality Commission's next inspection is in three months,"* Dr. Patel announced. *"We need a modern solution—one that can handle multiple consultants working simultaneously, process patients as they arrive in real time, and adapt to our dynamic A&E department where new patients walk in every few minutes."*

She turned to the hospital trust's IT contractor, **you**.

Royal Manchester Hospital operates 24/7 with:

- **Multiple specialised consultants** working simultaneously (Paediatricians, Surgeons, Cardiologists) That work on a shift basis. Three specialists work a 12 hour shift after the next 3 work the corresponding 12 hour shift.
- **Continuous patient arrivals** throughout the day—unpredictable and random
- **Urgent need for efficiency**—NHS targets require 95% of patients seen within 4 hours
- **Complex matching requirements**—consultants can only see patients needing their speciality

### **Continuous Patient Arrivals (10 Marks)**

- Patients arrive randomly throughout the day and night
- System must handle arrivals whilst consultants are working
- Each patient requires a specific speciality (Paediatrician, Surgeon, Cardiologist)

### **Automated Simulated Shift Management (10 Marks)**

- **Day shift** \* 3 consultants (12-hour shift)
- **Night shift** \* 3 consultants (12-hour shift)
- Automatic rotation between shifts

- Smooth handover without losing patients

### **Concurrent Processing (10 Marks)**

- Multiple consultants work simultaneously
- Each consultant treats patients matching their speciality
- Thread safe patient queue management
- No data corruption during concurrent access

### **Vodcast: Total 40 Marks**

Justify your choice of mechanisms on why it is most suitable for this scenario elaborating why it is better than other similar mechanisms. Justification will be done in a short Vodcast where you will:

1. Execute your solutions demonstrating your work with appropriate outputs clearly displayed **[10 Marks]**
2. **Explain your code with justifications** for your chosen concurrency control mechanisms to manage the concurrency issue of the requirements. Discussions may among others centre on concurrency, resource sharing, safety properties, liveness properties. It is important your explanations are technical and much beyond surface explanations as marks will be reduced in each section if explanations use language not at the level expected for Level 6 (Year 3 students).

**Code understanding:** The Student should be visible on screen during the vodcast during the vodcast and should not read off a script **[30 Marks]** Vodcast should be approximately 10 mins long. Failure to be visible on screen, or just reading a script, you forfeit all vodcast marks. Similarly, the quality and depth of your explanations should go beyond surface level for example simply saying “here is a thread and its running concurrently with other threads, booking appointments” is not a technical explanation and will lose marks for the Vodcast and marks for the code will be trimmed even if the solution works as expected.

3. **If a vodcast is not submitted the overall grade is capped at 40%**

### **Components to Submit**

- (1) The Java source code for each scenario **Note** submit a zip folder with all 2 projects scenarios in an easily runnable state. That is your marker should not have to make changes to your code to make it run. Submit a full project structure from Netbeans IDE or an agreed IDE with your module leader.  
**[60 MARKS]**
- (2) A vodcast demonstrating your applications and the output produced. Walk through the process giving detailed appropriately technical explanation and justification of your

chosen concurrent mechanism to control access to the resource. Offer some justifications as to why the chosen concurrency mechanism

**[40 MARKS]**

NOTE: We reserve the right to invite you in for a viva on your coursework. Failure to attend can result in a fail grade.

### **Viva Examination Requirement:**

The viva examination constitutes a critical component of this coursework assessment. Marks will be allocated primarily based on the student's individual performance and ability to demonstrate a clear understanding of their submitted work during the viva. **Failure to adequately justify or explain the submission, or inability to respond appropriately to viva questions, may result in a substantial reduction of marks or a fail grade for the module, irrespective of the quality of the submitted report or implementation.**