

Assessment Brief

| | | |
|---|---|--------------------------------------|
| Module Leader: Chris Bates | | Level:6 |
| Module Name: Designing and Developing Enterprise Systems | | Module Code: 55-609995-AF |
| Assignment Title: Prototype implementation | | |
| Group | Weighting: 70% | Magnitude: Equivalent to 3,500 words |
| Submission date/time: 5 th December, 2024 at 15:00 | Blackboard submission: Yes Turnitin submission: No | Format: Source code and video |
| Planned feedback date: 4 th January, 2025 | Mode of feedback: Written via Blackboard | In-module retrieval available: No |
| In this assessment are students asked to consider: | Inclusivity and accessibility | Not applicable |
| | Sustainability | Yes |

Module Learning Outcomes

- Synthesise how to critically assess the architectures of popular frameworks and the role which such frameworks have in modern software systems development is to underpin your competence for key areas of your practice, with a focus on your development of the following occupational Knowledge areas identified in your Training Plan.
- Assemble and employ skills through the construction of knowledge and techniques within the module and in your job role, to demonstrate and evaluate your existing and potential impact through the following occupational skills areas identified in your Training Plan.
- Construct experiential review and synthesise how by showing a strong work ethic and commitment to meet the required standards in relation to your personal development and your individual interpretation of reliability, working independently and within a team that suit your Professional Behaviours required by your occupational role and identified in your Training Plan.
- To apply evaluative reflective practice including synoptic review and action planning to summarise progress on the themes of distributed applications concepts, design and evaluation of frameworks and their relationships. Including types of patterns and large-scale application using, for example, services and microservices

Introduction

In this task you will design, build, and test a small distributed application. You can use any suitable language and platform. This is **not** a Web module and you should try to avoid the use of a Web server and Web browser if possible. If you only know Web development, consider whether your preferred language can be used to build other types of application (you can build different servers on nodejs, for example). You can use technologies associated with the

Web such as HTTP and JSON. A Web application can meet the pass criteria provided it demonstrates client pull, server push, and persistent storage.

This is a complicated problem and so you will work in groups of three or four. Try to work with people from different organisations so that you get to experience different ideas about software development.

The Task

In this task you will build a simulation of a smart meter system for recording and managing domestic electricity consumption.

You must build a client that acts as a smart meter. It should send readings to the server where they will be gathered, verified, and stored. The server will calculate a bill based on the current electricity price which it will return to the client. The client will display the bill. The client will send meter readings at random times but spaced no further than 60 seconds apart and no closer together than 15 seconds. You will demonstrate your system with at least twelve clients connected concurrently to the server.

The client must run without any input from a user and must be able to:

- Authenticate onto the server with a secure connection.
- Generate realistic randomly spaced readings.
- Send reading to the server.
- Display error messages if there is a communication problem between it and the server.
- Display a live bill that updates when each reading is transmitted.

The server must:

- Support multiple concurrent clients.
- Receive readings from each client.
- Store client details and readings.
- Calculate live bills and push them to clients.
- Log error messages if there is a communication problem between it and the server
- Push alerts to clients if there is a problem with the electricity grid.

Notes:

- You do not need to use a database. You may mock the database if you wish. When your system starts each client should be associated with an existing set of readings.
- Your client can be a command line application a – it does not need to be a Web or desktop application.,
- Messages should be formatted using XML, JSON, or YAML.
- You must use a version control system such as Github. You will be asked to give a link to your shared Github repository and we will expect to see commits made by all members of the team.

You must create a screencast in which you demonstrate your application, discuss its architecture, and show us the implementation of key features of both the server and the client. This video should last no longer than 30 minutes. All members of the team must contribute equally to the video.

Submission

You must submit a document that contains public links to your Github repository that holds your code and to your explanatory video to the Blackboard assignment handler by **3 p.m. on Thursday, 5th December, 2024.**

The code repository must include your source code and build instructions so that we can compile and run your code should we need to. Each of you **must** submit a link to the repository.

You **do not** need to upload your walkthrough video to Blackboard. Store it in your student Google drive space and provide a sharing link in your submission.

Group work

We recognize that group work can be challenging. The final day of the block week is your opportunity to start to work as a team. You should use this time to design and rapidly prototype your solution. You should use appropriate project management tools including a Github repository, and Teams, or similar, group chat.

If you are unable to work in a group or your group becomes dysfunctional you must talk to one of the tutors.

Grading criteria

Your work will be graded using the University's Grading Descriptor which is attached at the end of this task specification.

| | Marks | At a pass level you will... | At a first class level you will... |
|--|-------------------|--|--|
| The solution | Total of 80 marks | | |
| Completeness of the server solution | 20 | <ul style="list-style-type: none">• Implement some of the functionality of the backend. The solution is incomplete in key areas, but the parts that you. have built work properly.• Build a Web server-based application.• Use a simple database. | <ul style="list-style-type: none">• Provide implementations of all of the required functionality.• Built a robust server.• Implemented some error handling.• Used mocking instead of a database. |
| Completeness of the client | 20 | <ul style="list-style-type: none">• Have a basic client.• Have implemented some aspects of the system in the user interface.• Have a client that is a Web page.• Shown little consideration of user design or usability.• Write poor quality code. | <ul style="list-style-type: none">• Create a functionally complete client application.• Consider the usability of your application.• Write applications that are fully integrated with the back-end services.• Have produced high quality code. |
| Choice of technologies | 10 | <ul style="list-style-type: none">• Used Web technologies.• You build a simple command-line application. | <ul style="list-style-type: none">• Chosen an appropriate server language.• Choose a front-end technology that is suitable for platform-independent desktop UIs.• Use testing and mocking frameworks. |
| Architecture and patterns | 20 | <ul style="list-style-type: none">• Demonstrated limited use of a standard architecture.• Used only basic patterns such as model-view-controller. | <ul style="list-style-type: none">• Have a clear and appropriate architecture.• Show that you can use software design patterns appropriately. |
| The use of specifications and unit testing | 10 | <ul style="list-style-type: none">• Write a few unit tests.• Attempt to use mocks. | <ul style="list-style-type: none">• Have a well-structured approach to mocking the storage of data.• Have good test coverage of the back end.• Write tests that exercise a variety of aspects of the server. |

| The video | Total of 20 marks | | |
|--|-------------------|--|--|
| Demonstration and explanation of your code | 20 | <ul style="list-style-type: none"> • Create a video which shows you using the program. • Show little of the code • Give explanations that are weak, incomplete, or incorrect. | <ul style="list-style-type: none"> • Make a video in which you clearly show the program being used. • Show and explain important or interesting sections of code. • Explain how the code works and why you wrote it as you did. |

Artificial Intelligence and Academic Integrity

It is important you do not use AI tools to generate an assignment and submit it as if it were your own work. Our regulations state:

Contract cheating/concerns over authorship: This form of misconduct involves another person (or artificial intelligence) creating the assignment which you then submit as your own. Examples of this sort of misconduct include buying an assignment from an ‘essay mill’ or professional writer; submitting an assignment which you have downloaded from a file-sharing site; acquiring an essay from another student or family member and submitting it as your own; attempting to pass off work created by artificial intelligence as your own. These activities show a clear intention to deceive the marker and are treated as misconduct.

Further guidance is available here: <https://blogs.shu.ac.uk/assessment4students/preparing-to-submit-work/#AI>

Level 6 - Generic grade descriptor: relationship of degree classification and equivalent percentage

| Class | Category | Mark range | % | General Characteristics |
|-----------------|-----------------|------------|----|---|
| 1 st | Exceptional 1st | 93 - 100 | 96 | Exceptional breadth and depth of knowledge and understanding of the area of study, significantly beyond what has been taught in all areas ; evidence of extensive and appropriate selection and critical evaluation/synthesis/analysis and of reading/research beyond the prescribed range, in both breadth and depth, to advance work/direct arguments; excellent communication; performance deemed to be beyond expectation. Work at publishable or commercial standard. The ability to make decisions and systematically carry out tasks/processes with autonomy in unpredictable situations; exercise of initiative in the completion of practical tasks; exceptional leadership skills and evidence of personal responsibility in group contexts; creative flair; extremely well-developed problem-solving skills; the ability to carry out sustained critical reflection on practical work within the wider context of industry/workplace. Exceeds expectations set by the industry/employment context. |
| 1 st | High 1st | 85 - 92 | 89 | Excellent knowledge and understanding of the area of study as the student is typically able to go beyond what has been taught (particularly for a mid/high 1st) ; evidence of extensive and appropriate selection and critical evaluation/synthesis/analysis of reading/research beyond the prescribed range, to advance work/direct arguments; excellent communication; performance deemed beyond expectation of the level. The ability to make decisions and carry out tasks/processes with a high level of autonomy; creative flair and the ability to (re)interpret predefined rules/conventions to select and justify individual working practice; excellent problem-solving skills; accuracy and fluency; excellent command of skills appropriate to the task; the ability to reflect critically on practical work within the wider context of industry/workplace. Meets expectations set by the industry/employment context. |
| | Mid 1st | 78 - 84 | 81 | |
| | Low 1st | 70 - 77 | 74 | |
| 2.1 | High 2.1 | 67 - 69 | 68 | Very good knowledge and understanding of the area of study as the student is typically able to relate facts/concepts together with some ability to apply to known/taught contexts ; evidence of appropriate selection and critical evaluation of reading/research, some beyond the prescribed range, may rely on set sources to advance work/direct arguments; demonstrates autonomy in approach to learning; strong communication skills. Broadly autonomous completion of practical tasks/processes; ability to adapt in response to change or unexpected experiences; technical/artistic decision making is very highly developed; a clear command of the skills relevant to the task/process; ability to reflect on practical work and set future goals within the wider context of industry/workplace. Adherence to standards set by the industry/employment context. |
| | Mid 2.1 | 64 - 66 | 65 | |
| | Low 2.1 | 60 - 63 | 62 | |
| 2.2 | High 2.2 | 57 - 59 | 58 | Good knowledge and understanding of the area of study balanced towards the descriptive rather than critical or analytical ; evidence of appropriate selection and evaluation of reading/research, some may be beyond the prescribed range, but generally reliant on set sources to advance work/direct arguments; communication shows clarity, but structure may not always be coherent. A confident approach to practical tasks; a solid grasp of the related processes, tools, technology; creativity in the completion of the task; proficiency is demonstrated by an accurate and well-coordinated performance; tasks are completed with a good level of independent thought and autonomy; an ability to reflect on practical work and set future goals. General adherence to standards set by the industry/employment context. |
| | Mid 2.2 | 54 - 56 | 55 | |
| | Low 2.2 | 50 - 53 | 52 | |
| 3 rd | High 3rd | 47 - 49 | 48 | Knowledge and understanding sufficient to deal with terminology, basic facts and concepts but fails to make meaningful synthesis; some ability to select and evaluate reading/research however work may be more generally descriptive; general reliance on set sources to advance work; arguments may be weak or poorly constructed; communication/presentation is generally competent but with some weaknesses. Competence in technical/artistic skills; tasks/processes are completed with a degree of proficiency and confidence; tasks are completed with a sufficient level of independent thought; effective judgements have been made; evaluation and analysis of performance in practical tasks is evident. Errors in workflow or completion of the task; general adherence to appropriate rules/conventions set by the industry/employment context. |
| | Mid 3rd | 44 - 46 | 45 | |
| | Low 3rd | 40 - 43 | 40 | |
| Fail | Borderline Fail | 30 - 39 | 35 | Insufficient knowledge and understanding of the subject and its underlying concepts; some ability to evaluate given reading/research however work is more generally descriptive; naively follows or may ignore set material in development of work ; given brief may be only tangentially addressed or may ignore key aspects of the brief; communication shows limited clarity, poor presentation, structure may not be coherent. Practical tasks are attempted; skill displayed in some areas; there are a significant number of errors; a lack of proficiency in most areas; guidance may be needed to reproduce aspects of the task and/or apply learned skills. Tasks may be incomplete; failure to adhere to some of the rules/conventions set by the industry/employment context. |
| | Mid Fail | 20 - 29 | 25 | |
| | Low Fail | 6-19 | 10 | No evidence of knowledge or understanding of the subject; no understanding of taught concepts, with facts being reproduced in a disjointed or decontextualised manner ; ignores set material in development of work; fails to address the requirements of the brief; lacks basic communication skills. A general level of incompetency in practical tasks; an evident lack of practice; set tasks are not completed; few or no skills relating to tasks are evident. No adherence to rules/conventions set by the industry/employment context. |
| Zero | Zero | 0-5 | 0 | Work not submitted, work of no merit, penalty in some misconduct cases. |