

Assessment Brief



Subject Code and Name	DEV1002 - Databases & Servers
Assessment Number	3
Assessment Title	Web API Server
Assessment Type	Individual Programming Project
Words, Size or Duration	1 code project
Subject Learning Outcomes	SLO3, SLO4, SLO5, SLO6
Submission Date / Time	Due by 11:55pm AEST Sunday end of Module 4.
Weighting	45%

Assessment Purpose

Web servers can come in many shapes & contain different levels of complexity. At their core, they always involve server concepts such as routing & handling the communication of data between users & a data storage medium.

To solidify your knowledge of core web server concepts and show your ability to work with web servers at a fundamental level, you should be able to write code to create a functioning web API server.

Assessment Task / Item

For this assessment, you must submit an application that meets the design & programming requirements to showcase your skills as a software developer.

Assessment Brief

Assessment Instructions

Design a functioning web server in the relevant programming language. Your web server must contain valid, functioning code and adhere to the following requirements:

Project requirements:

- Complete a planning stage before developing the application, which requires the development of these items:
 - An entity relationship diagram (ERD) that represents the database & data structures planned to be used with the web server
 - An explanation of the chosen database system, including comparisons to other types of database systems
- Seek feedback from at least two others, and describe how you appropriately responded to feedback gathered from the planning stage, then provide a justification for your response.

Design requirements:

- The web server must:
 - be deployed to a publicly-accessible URL
 - function properly in both local & deployed environments
 - store data in a persistent data storage medium (eg. a relational or non-relational database)
 - appropriately validate & sanitise any data it interacts with
 - use appropriate HTTP web request verbs for various types of data manipulation
 - cover the full range of CRUD functionality for data within the database
- An entity relationship diagram must be created to represent the database.
- The database manipulated by the web server must accurately reflect the entity relationship diagram.
- The database tables or documents must be normalised to third normal form (3NF) or higher, or other suitably-complex & optimised normalisation forms.

Programming requirements:

- The web server must:
 - use appropriate functionalities or libraries from the relevant programming language in its construction
 - handle errors gracefully, not crashing or breaking when things like invalid user input occur
 - use appropriate error-handling techniques to catch errors
 - use appropriate functions or methods to sanitise & validate data
 - use D.R.Y coding principles

Assessment Brief



Submission

All work must be submitted via Canvas, in the assignments section appropriate to this brief. Please ensure the above mentioned submission date and/or time are adhered to, or penalties may apply.

When submitting your work, please save your files using the naming convention below.

[Student_ID]_[Surname]_[First Name]_[SubjectCode]_[Assessment_#].zip

E.g.: **1234_Singh_Visha_PRG1002_Assessment_01.zip**

For more information on late submission, please see the **Assessment Policy**.

Academic Integrity

The integrity of the assessment process is fundamental for ensuring appropriate evaluation at AIT. All work submitted should be your own, and where additional resources are used, they must be referenced according to the Harvard style. Additionally, TurnItIn is available in the LMS to test plagiarism in your writing.

For more information on academic integrity, please see the **Academic Integrity** and **Academic Integrity Penalties Policies**.

Appeals

Fair application of the assessment rubric, rules and guidelines should be administered for each assessment. If you feel an evaluation requires further consideration, you may be entitled to an appeal.

For more information on your right to an appeal, please see the **Assessment Appeals Procedure and Policy**.

Policies

For access to the policies mentioned above and related to education at AIT, please see the [footer](#) of the AIT website, and follow the link named **Education Policies and Procedures**.

Website: <https://www.ait.edu.au>

Assessment Brief



Assessment Rubric

Task Descriptor	(HD) High Distinction (85-100%)	(D) Distinction (75-84%)	(C) Credit (65-74%)	(P) Pass (50-64%)	(F) Fail (0-49%)
<i>CREATES an application which handles errors.</i> 10% SLO 4, SLO 6	Application handles ALL CATEGORIES of errors GRACEFULLY.	Application handles MOST CATEGORIES of errors GRACEFULLY.	Application handles FEW CATEGORIES of errors GRACEFULLY.	Application handles ONE CATEGORY of errors GRACEFULLY.	Does not create an application which handles any category of errors.
<i>APPLIES DRY (Don't Repeat Yourself) coding principles to code.</i> 5% SLO 6	Applies DRY coding principles to ENTIRE codebase.	Applies DRY coding principles to MOST OF code base.	Applies DRY coding principles IN SEVERAL PLACES within the code base.	Applies DRY coding principles IN FEW PLACES within the codebase IN A BASIC WAY.	Does not apply DRY coding principles.
<i>APPLIES programming language features or frameworks to manipulate a data model.</i> 15% SLO3, SLO 4, SLO 6	Very well-written code that manipulates the data model with full CRUD functionality.	Well-written code that manipulates the data model with full CRUD functionality.	Code manipulates the data model with almost full CRUD functionality.	Code manipulates the data model at least two CRUD functionalities.	Does not implement any working CRUD functionalities, or implements only one.
<i>IMPLEMENTS input validation and integrity checks on data to address business risks.</i> 10% SLO 4, SLO 6	Completely correctly checks for null values and provides error checking to ensure integrity of information in the database.	Mostly correctly checks for null values and provides error checking to ensure integrity of information in the database.	Somewhat correctly checks for null values and provides error checking to ensure integrity of information in the database.	Marginally correctly checks for null values and provides error checking to ensure integrity of information in the database.	Does not implement any working validation or integrity checks on data.
<i>DEVELOPS suitable entity relationship diagrams to represent the data model</i> 10% SLO 5	Diagram provided represents ALL entities intended for the data model with COMPLETELY ACCURATE relationships depicted, using BASIC & COMPLEX relationships where appropriate.	Diagram provided represents ALMOST ALL entities intended for the data model with MOSTLY ACCURATE relationships depicted, using BASIC & COMPLEX relationships where appropriate.	Diagram provided represents MOST entities intended for the data model with MOSTLY-ACCURATE relationships depicted, using BASIC relationships where appropriate.	Diagram provided represents SOME entities intended for the data model with MOSTLY-ACCURATE relationships depicted, using BASIC relationships where appropriate.	Diagram either not provided, not depicting any relevant entities, or not depicting any relevant entity relationships.



Assessment Brief

<i>OPTIMIZES an entity relationship diagram to a standard format.</i> 10% SLO 5	ERD depicts database normalisation at 3NF or another suitably-complex normalisation form such as Boyce-Codd or 4NF.	ERD depicts database normalisation mostly in 3NF, with some entities still depicting 2NF.	ERD depicts database normalisation in 2NF, and may include some entities still depicting 1NF.	ERD depicts database normalisation in 1NF.	ERD does not depict or attempt any database normalisation, or the ERD represents data at 0NF.
<i>IMPLEMENTS database normalisation in an appropriate scripting language.</i> 10% SLO 4, SLO 5, SLO 6	Implemented code represents & works with data at 3NF, or another suitably-complex normalisation form such as Boyce-Codd or 4NF.	Implemented code represents & works with data at 3NF, with few entities using 2NF.	Implemented code represents & works with data at 2NF, with few entities using 1NF.	Implemented code represents & works with data at 1NF.	Implemented code does not match the project's ERD, or otherwise does not implement database normalisation to any practical level.
<i>DEVELOPS a web server that conforms to standard communication protocols.</i> 10% SLO 4	COMPLETELY APPROPRIATE usage of HTTP verbs, with NUMEROUS verbs implemented within the web server code appropriate for CRUD functionality.	COMPLETELY APPROPRIATE usage of HTTP verbs, with ONE OR TWO verbs implemented within the web server code appropriate for CRUD functionality.	MOSTLY APPROPRIATE usage of HTTP verbs, with NUMEROUS verbs implemented within the web server code appropriate for CRUD functionality.	MOSTLY APPROPRIATE usage of HTTP verbs, with ONE OR TWO verbs implemented within the web server code appropriate for CRUD functionality.	Implementation of HTTP verbs is either non-functional, not attempted, or otherwise not appropriate for a web server.
<i>DEPLOYS a web server to an online hosting service.</i> 10% SLO 4	SUCCESSFULLY deploys to a cloud hosting service, uses environment variables, uses the same database type in a development/testing environment as production, uses custom domain name.	SUCCESSFULLY deploys to a cloud hosting service, uses environment variables, uses the same database type in a development/testing environment as production.	SUCCESSFULLY deploys to a cloud hosting service, and the server uses environment variables.	SUCCESSFULLY deploys to a cloud hosting service and the server MOSTLY functions in a deployed production environment.	App is NOT deployed successfully to a cloud hosting service or DOES NOT function in a deployed production environment.
<i>ADAPTS to feedback throughout the planning and development of the application.</i> 10% SLO 6	APPROPRIATELY implements ALL feedback gathered at MULTIPLE stages during the project.	APPROPRIATELY implements MOST feedback gathered at MULTIPLE stages during the project.	APPROPRIATELY implements MOST feedback gathered at ONE stage during the project.	Implements SOME feedback gathered at ONE stage during the project.	Does NOT implement ANY feedback gathered during the project.