ASSESSMENT AND INTERNAL VERIFICATION FRONT SHEET (Individual Criteria)

Course Title	**B.Sc Software Development (Part Time)**		**Lecturer**	**Ryan Attard**	
Unit Number & Title	**ITSFT-506-2012-Securing Applications**				
Assignment Number, Title / Type	**Developing a Secure Web Application**				
Date Set		**Deadline Date**	**May 10 2021**		
Student Name		**ID Number**		**Class**	

Assessment Criteria		Maximum Mark	Mark Achieved
KU5	Solve a problem of authenticity of data by selecting a cryptographic technique	5	
KU6	Solve a problem of integrity of data by selecting a proper cryptographic technique	5	
AA1	Produce a solution that mitigates against XXE and XSS	7	
AA2	Produce a solution that mitigates against injection (sql, command, file, etc)	7	
AA3	Employ monitoring and logging while arranging various logs and error logs such that handling of these is done in a secure way	7	
AA4	Apply encryption techniques to hide sensitive information	7	
SE1	Analyze and employ a strong authentication and authorization techniques against bypasses, directory traversals, etc	10	
SE2	Assess the shortcomings in a given scenario of using a simple cryptographic technique and develop a solution	10	
SE3	Compose a post assessment testing report which should draw conclusions about the security of a given scenario/application	10	

Notes to Students:			
Total Mark		68	

- This assignment brief has been approved and released by the Internal Verifier through Classter.

- Assessment marks and feedback by the lecturer will be available online via Classter (Http://mcast.classter.com) following release by the Internal Verifier

- Students submitting their assignment on Moodle/Unicheck will be requested to confirm online the following statements:

Student's declaration prior tohanding-in of assignment

- ❖ I certify that the work submitted for this assignment is my own and that I have read and understood the respective Plagiarism Policy

Student's declaration on assessment special arrangements

- ❖ I certify that adequate support was given to me during the assignment through the Institute and/or the Inclusive EducationUnit.
- ❖ I declare that I refused the special support offered by theInstitute.

Assignment

Securing Applications

Assessors: **Ryan Attard**
Assessment Type: **Home assignment**

Assignment Guidelines

Read the following instructions carefully before you start the assignment. If you do not understand any of them, ask your invigilator.

- This assignment is a HOME assignment.
- Fill in and print the assignment sheet and produce a properly structured, neat documentation.
- Copying is **Strictly Prohibited** and will be penalised according to disciplinary procedures.
- Use the given cloud account responsibly
- Deadline: _____
- This assignment has a total of 68 marks.
- Submission must be done through Moodle
    - Zipped code must be supplied/ Git repository link
    - Document/Text containing the public link to your website (if you managed to answer the relevant task)

Develop a website which should be used by students and teachers to exchange "Tasks" information between them. Summary of functionalities:

Teacher:

- Teachers should log in using 3rd Party login credentials
- Should be able to create accounts for students
    - Details of account should be automatically sent via email, including a randomized password, without the teacher knowing what the password is.
- Create a new Task (with description and deadline)
    - Deadline should be validated and respected
- Ability to view and download/open data submitted by students, whilst identifying who that data belongs to.
    - Further details in no. 1
- Ability to comment on the uploaded work. This can be done as a comment on the uploaded work, which a student can reply to via his/her commenting feature.

Student

- Should be able to log in, with the user account created by the Teacher, while then being able to set up 2fa.
- Ability to view a list of Tasks created by the teacher that created his account, including the deadline and a page from where to upload his work related to the Task.
- Ability to submit his/her work in the page provided related to a given Task.
    - Students can submit files.
        - Further details in no. 1
    - Files allowed are .pdf, which can either then be downloaded by teacher to assess or open them in the browser itself.
    - Files submitted should be protected (privacy, authenticity and access to the resource)
    - Once deadline is elapsed, student cannot upload the work.
- Students can comment on their workBUT input should be handled securely . Use ActionFilter.
- Students can view ONLY their work whenever they want. Use ActionFilter.

Further Details

1. Uploaded work should be verified for authenticity and integrity. Therefore uploaded files should be digitally signed and when they are about to be viewed by the lecturer/student
    a. They should be first verified if an identical copy was already uploaded by someone else (in which case a flag is displayed on screen to the teacher indicating that uploaded work is copied)
    b. And when the teacher is about to download the file, you digitally verify it and so therefore whether their authenticity still holds or not
    If any of the above fails a notification should be displayed.
2. File Access Logs and Error Logs should be kept in a safe place(in a text file not accessible from the outside) and implementations should be done in a recommended way. No errors should be allowed to disclose any information to the end user.

Comment [RA1]: Student can comment on their work ONLY

Comment [RA2]: In the table/ model of the class containing file details you must also have a column called signature

Comment [RA3]: Choose the safe place wisely

3. Choose any of the above features and code it (partially or in full) using some client side code. Obfuscate that code while making sure that the way its called and the data that is passed to the server is handled securely

KU5 Solve a problem of authenticity of data by selecting a cryptographic technique

- ❒ Before assessing the submitted work, it should be first checked for authenticity

Comment [RA4]: Authenticity: this refers to digital signing and digital verification

KU6 Solve a problem of integrity of data by selecting a proper cryptographic technique

- ❒ Before assessing the submitted work, it should be first checked for integrity

Comment [RA5]: Hashing (is always one way)

AA1 Produce a solution that mitigates against XXE and XSS

- ❒ Any input should be validated properly and made sure that XSS execution is not allowed. Failure to mitigate any critical input will result in a deduction of 1pt everytime. Note: In the text box students/teachers can write what is considered to be malicious code, however it should be never rendered/executed therefore it should be handled securely[2]
- ❒ Permission (for student) to comment on their own work only; Teachers can comment on anyone's work; Both should be validated using ActionFilter [2]
- ❒ FileAccess should be restricted only to the owner of the file and the teacher. Restrict all physical file access; should be validated using ActionFilter [2]
- ❒ Deadlines should be properly checked when material is uploaded by the student [1]

Comment [RA6]:

Comment [RA7]: Student A cannot comment on student b's work
Student A cannot download student b's work

AA4 Apply encryption techniques to hide sensitive information

- ❒ Details of account should be automatically sent via email, including a randomized password, without the teacher knowing what the password is. Therefore passwords should also be hashed in the db (2)
- ❒ Querystrings should be encrypted and decrypted [3]
- ❒ Shows evidence of code obfuscation [2]

Comment [RA8]: Both encryption and decryption must be working

AA2 Produce a solution that mitigates against injection (sql, command, file, etc)

- ❒ Implement your Business Logic to be safe against SQL Injection (3.5)
- ❒ Make sure your application is not vulnerable against file Injection (3.5)

AA3	Employ monitoring and logging while arranging various logs and error logs such that handling of these is done in a secure way

- ❒ Error Logs [2]
- ❒ File Access Logs including info such as (ip address, timestamp, user, etc) [3]
- ❒ Error Pages are used in a proper way [2]

SE1	Analyze and employ a strong authentication and authorization techniques against bypasses, directory traversals, etc

- ❒ Roles Authorization should be respected and used in all services provided [2]
- ❒ Teachers should use a Google/Microsoft account [3]
- ❒ Students should use a Custom Forms login, with details generated by the teacher with 2FA QR SCAN image enabled [3]
- ❒ Protection against CSRF should be included everywhere [2]

Comment [RA9]: On the post methods

SE2	Assess the shortcomings in a given scenario of using a simple cryptographic technique and develop a solution

- ❒ Files should be encrypted using Hybrid encryption when uploaded and decrypted when opened by the teacher/student.

Comment [RA10]: Both encryption and decryption must be working to get 10 marks

Testing/ Attacking

1. Make use of any tools which you are familiar with and try to test and explore any of these vulnerabilities
    a. Broken Authentication
    b. Sensitive Data Exposure
    c. Broken Access Control
    d. Injection

2. After that compile a detailed report (**worthy of 10 marks**) and explain in detail what testing you have carried out in (1). Report must clearly show and explain (in step-by-step if necessary) what tools and how did you employ these tools, while also describing what you tested and what conclusions you can draw after this thorough testing.

SE3	Compose a post assessment testing report which should draw conclusions about the security of a given scenario/application

- ❒ Report must clearly show the usage of at least 1 tool but at least 3 different features
- ❒ Report must show what assets and threats simulated to attack the aforementioned assets.Threatmodelling templates should be followed[1].
- ❒ Report must be technical, showing screenshots of tests carried out, and any rationale behind carrying out such tests
- ❒ Report must draw conclusions with regards to implemented mitigations outlining any possible shortcomings (outlining any possible improvements which you could apply) – this will in no way have any effect on your mark as long as you followed all the instructions above.

Comment [RA11]: Minimum of 3 pages describing what you are doing
And 1 page for the assets + threats

Comment [RA12]: 3 threats i.e 3 tables showing details of threats

Comment [RA13]: Screenshot of the software and the data it shows indicating what you are trying to describe

[1]See appendix

- ❒ If all the tests ran, were (not) successful (i.e. all attacks mitigated), you should show "what could have happened if…" so you clearly show that the tool you used is really working right.

Failure to document any of the above will penalize 2 points every time.

Appendix A - Templates

Trust Levels:

ID	Name	Description
T1	Remote anonymous user	A user who has not yet authenticated to the website
T2	Authenticated user	A registered user who has valid credentials
T3	Database server administrator	User who can do any operation on the underlying database
T4	Website administrator	A user who can configure the website by for example uploading new website versions, deleting files from the web server etc.
T5	Clerk	A user having this role is able to maintain products, their prices, categories and sub categories.

Assets:

ID	Name	Description	Trust level
A1	User	Assets that relate to a website user	
A1.1	User's login data	User's credentials username and password. This asset needs protection because if it is stolen another user would be able to do anything which the user can do	T2 Authenticated user T3 Database server admin

A1.2	User's personal data	User's personal data including contact information. This needs protection because some personal data might be important such as telephone number	T2 Authenticated user T3 Database server admin

Threats:

Id	TR1
Name	Adversary tries to supply malicious data when logging in
Description	Adversary tries to input special characters to be able pose as another user, or logs in without having an appropriate username and password. Handling of data is critical in this regards.
Stride	Tampering, Elevation of privilege
Entry Points	(E1.1) Login page
Assets	(A1.2) User's personal data, (A2) Backend database
Mitigation Strategy	Using stored procedures or parameterized queries