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**	
Student Name		**ID Number**	**Class**

☐	*Student's declaration prior to handing-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 (Tick only if applicable)*** ❖ *I certify that adequate support was given to me during the assignment through the Institute and/or the Inclusive Education Unit.* ❖ *I declare that I refused the special support offered by the Institute.*

Student Signature:		**Date :**	

Assessment Criteria		Maximum Mark	Mark Achieved
KU5	Solve a problem of authenticity of data by selecting a cryptographic technique		
KU6	Solve a problem of integrity of data by selecting a proper cryptographic technique		
AA1	Produce a solution that mitigates against XXE and XSS		
AA2	Produce a solution that mitigates against injection (sql, command, file, etc)		
AA3	Employ monitoring and logging while arranging various logs and error logs such that handling of these is done in a secure way		
AA4	Apply encryption techniques to hide sensitive information		
AA5	Make use of third-party tools/ scanners to help in the identification of vulnerabilities		
SE1	Analyze and employ a strong authentication and authorization techniques against bypasses, directory traversals, etc		
SE2	Assess the shortcomings in a given scenario of using a simple cryptographic technique and develop a solution		
SE3	Compose a post assessment testing report which should draw conclusions about the security of a given scenario/application		
Total Mark			

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 75 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 no. 1
- Ability to comment on the uploaded work.

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 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
- Students can comment on their work
- Students can view ONLY their work whenever they want.

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
    a. They should be first verified if a similar copy was already uploaded by someone else
    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 and implementations should be done in a recommended way. No errors should be allowed to disclose any information to the end user.
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

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

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 (e.g. deadlines) will result in a deduction of 2pts every time. Note: In the text box students/teachers can write what is considered to be malicious code, however it should be never rendered/executed [2]
- ❒ Permission to comment on their work only [2]
- ❒ FileAccess should be restricted only to the owner of the file and the teacher. Restrict all physical file access [3]

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]

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
- ❒ Teachers should use a Google/Microsoft account
- ❒ Students should use a Custom Forms login, with details generated by the teacher with 2FA

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.

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.

AA5	Make use of third-party tools/ scanners to help in the identification of vulnerabilities

- ❒ This will be assessed during the interview, where the interviewer will ask the interviewee how to use the tool for a specific “test” or “attack”

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.
- ❒ 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)
- ❒ 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.