**The Ministry of Education of the Azerbaijan Republic**

**The State Oil Company of the Azerbaijan Republic**

**Baku Higher Oil School**

Information Technology Department

Major: Information Security (Bachelor degree)

**Web Programming and Security**

**Courses Syllabus**

Fall, 2024

Instructor : Lec. Elnur Baghirov

Course code: IT-22.1 Course credit : 9.08

Office : 410, Campus Aypara Office hours : S 09.00-14.20

Prerequisites: -

Language of instruction: English

Schedule :

* IT-22.1: Lecture
* IT-22.1: Laboratory

Web site : in progress

Email : [elnur.bagirov.std@bhos.edu.az](mailto:elnur.bagirov.std@bhos.edu.az)

**Description about course**

**Web Programming and Security** is a comprehensive course designed to introduce students to the development and security of web applications using Java. The course emphasizes both the practical and theoretical aspects of building secure, dynamic, and responsive web applications.

Students will learn essential web technologies such as HTML, CSS, and JavaScript for frontend development and dive into Java-based backend technologies like Servlets and JSP for creating dynamic web content. As security is a critical component of web applications, the course also focuses on identifying and mitigating web vulnerabilities, such as Cross-Site Scripting (XSS), SQL Injection, and Cross-Site Request Forgery (CSRF), using industry-standard practices.

Throughout the course, students will gain hands-on experience through lab assignments and projects that require them to develop secure web applications, interact with databases, and integrate security mechanisms such as authentication, session management, and HTTPS. Additionally, they will learn to use tools such as OWASP ZAP for security testing and understand key web security protocols, including SSL/TLS and JWT for API protection.

By the end of the course, students will have a solid foundation in both web development and security, equipping them with the skills needed to build and secure modern web applications in a professional setting.

**Course objective section**

By the end of this course, students will be able to:

1. **Develop Web Applications using Java:**  
   Master the fundamentals of web development using Java-based technologies like Servlets, JSP, and JDBC, enabling students to build dynamic, database-driven web applications.
2. **Understand Frontend and Backend Integration:**  
   Gain a comprehensive understanding of how the frontend (HTML, CSS, JavaScript) and backend (Java, Servlets) components work together to deliver full-stack web applications.
3. **Implement Security Best Practices:**  
   Apply secure coding techniques and industry best practices to safeguard web applications against common vulnerabilities, such as Cross-Site Scripting (XSS), SQL Injection, and Cross-Site Request Forgery (CSRF).
4. **Use Security Tools for Application Testing:**  
   Learn to use tools like OWASP ZAP and Burp Suite for web application vulnerability scanning and penetration testing to identify and address security flaws.
5. **Secure Data Transmission and Communication:**  
   Implement secure communication protocols (SSL/TLS) to ensure data integrity and confidentiality in web applications, and protect REST APIs using OAuth2 and JWT for secure authentication.
6. **Handle Authentication and Session Management:**  
   Design secure login systems, manage user sessions, and enforce strong access controls to prevent unauthorized access to web resources.
7. **Apply Secure Database Interaction Practices:**  
   Safely interact with databases using Java and JDBC, and prevent SQL injection attacks by implementing prepared statements and other secure database access techniques.
8. **Develop and Deploy a Secure Web Application:**  
   Build a full-fledged secure web application as a final project, incorporating all the security measures and development practices learned throughout the course.

**Learning outcomes section**

 Develop dynamic web applications using Java Servlets, JSP, and JavaScript.

 Understand and apply security best practices to protect web applications.

 Identify, exploit, and mitigate common web vulnerabilities.

 Perform security assessments on web applications using industry tools.

 Secure RESTful APIs using authentication techniques such as OAuth2 and JWT.

**Assessment methods**

The exams are done using computers. Some part of exam is written examination. All questions must be answered.

**Grading**

**Type Weight Date Exam minutes**

Final 40% TBA (to be announced) 120

Quiz 15% 6th week of the semester 100

Laboratory 25% one lab per week deadline

Midterm Project: 20%

**Resit exam grade is equal to Final exam grade and equals to 40%**

**NOTE:**

* **Total number of Quizes 1 or 2. Each quiz is graded separately. At the end of the term the average mark of the quizes is calculated.**
* **Total number of labs is twelve. Each lab is graded separately. At the end of the term the average mark of the labs is calculated.**
* **Activity grade is given individually to students. It is shown how active is student during the class.**

**Area grading scale**

A 91-100

B 81-90

C 71-80

D 61-70

F ≤ 60

**Rules**

**Exams**

In order to be excused from the exam, the student must contact the dean and the instructor before the exam. Excuse will not be granted for social activities such as trips, cruises and sporting events (unless you are participating). The exams will all be cumulative. Most of the questions on each exam will be taken from the chapters covered since the last exam.

But some will come from the earlier chapters. In general, the coverage will reflect the amount of the time spend in class on the different chapters.

**Withdrawal (pass / fail)**

This course strictly follows grading policy of the Information Technology Department. Thus, a student is normally expected to achieve a total mark (preexam score + exam score) of at least 61 to pass. In this case of failure, he/she will be referred or required to repeat the course the following term or year.

**Late policy**

Late assignment submissions won’t be accepted for grading. The grade for this assignment will be **zero**.

**Teaching resources**

Presentations : Web Programming and Security (in site: [www.lms.bhos.edu.az](http://www.lms.bhos.edu.az))

Textbook :

[1] "Core Servlets and JavaServer Pages" by Marty Hall.

[2] MDN Web Docs (online resource).

[3] Pro JSP by Apress.

[4] OWASP (online resource).

[5] The Web Application Hacker's Handbook" by Dafydd Stuttard.

[6] "RESTful Java with JAX-RS" by Bill Burke

For class presentations and discussions, the student should utilize journal and internet materials. Moreover, the course does not limit the use of learning materials available at BHOS library.

**Attendance**

The students are required to attend all classes as a part of their studies and those having legitimate reasons for absence (illness, family bereavement, etc.) are required to inform the instructor.

**Professionalism and Participation**

1. Attend class regularly, arrive on time, leave only when dismissed

2. Attend class with all materials required, be prepared to listen and work

3. Be well prepared for class, read all required materials, and complete all necessary preparation

4. Be attentive in class, take notes, contribute to discussion and ask intelligent questions

5. Demonstrate professional and respectful interpersonal relationships with peers and instructor: ATTITUDE COUNTS, AND whining is unacceptable

6. Take responsibility for your actions, and your results

**Plagiarism**

Honesty requires that any ideas or material taken from another source for written, visual, or oral use must be fully acknowledged. Offering the work of someone else as one’s own is plagiarism. The language or ideas thus taken from another may range from isolated formulas, images, sentences or paragraphs to entire articles copied from books, periodicals, speeches, or the writings and creations of other students. The offering of materials assembled or collected by others in the form of projects or collections without acknowledgment also is considered plagiarism. Any student who fails to give credit for ideas or materials taken from another course is guilty of plagiarism.

|  |  |  |
| --- | --- | --- |
| **Week** | **Topics** | **Textbook/Assignments** |
| 1-2 | **Introduction to Web Programming with Java and**  **Frontend Basics (HTML, CSS, JavaScript)**   * Overview of web architecture and protocols (HTTP/HTTPS). * Introduction to Java in web development (Servlets, JSP). * Overview of security challenges in web programming.   - Overview of HTML5 structure, CSS3 styling, and JavaScript basics.  - Interaction between frontend and backend (form submission, HTTP requests). | - "Core Servlets and JavaServer Pages" by Marty Hall, Chapter 1 (Introduction to Servlets).   * MDN Web Docs (online resource): Introduction to HTML, CSS, and JavaScript. * Pro JSP by Apress, Chapter 2 (JSP and Servlets: Working with Forms). |
| 3-4 | **Java Servlets and JSP for Dynamic Web Pages**   * Introduction to Java Servlets and JSP. * Handling HTTP requests (GET, POST) and session management. * Separation of concerns (JSP for views, Servlets for logic). | * "Core Servlets and JavaServer Pages" by Marty Hall, Chapters 2-4 (JSP and Servlet Communication). |
| 5 | **Introduction to Web Application Security**   * Overview of the OWASP Top 10 vulnerabilities. * Importance of input validation and output encoding. * Basic authentication and authorization in web applications. | **-**  OWASP Java Security Cheat Sheet (online resource).  - Pro JSP by Apress, Chapter 7 (Authentication and Session Handling). |
| 6-7 | **Cross-Site Scripting (XSS) and**  **SQL Injection and Secure Database Interaction**   * What is XSS and how it impacts web security. * Types of XSS: Reflected, Stored, and DOM-based. * Prevention techniques: Input validation, output encoding, Content Security Policy (CSP). * Introduction to databases (MySQL/PostgreSQL) and Java Database Connectivity (JDBC). * SQL injection attacks: How they work and their consequences. * Preventing SQL injection using prepared statements and ORM tools (Hibernate). | **-** "The Web Application Hacker's Handbook" by Dafydd Stuttard, Chapter 12 (Cross-Site Scripting).  - OWASP XSS Prevention Cheat Sheet.  - "Core Servlets and JavaServer Pages" by Marty Hall, Chapter 9 (JDBC and Database Access).  - OWASP SQL Injection Prevention Cheat Sheet. |
| 8-9 | **Secure Session Management and CSRF Protection,**  **Web Application Firewalls (WAF) and Security Tools**   * Session management in Java (using HttpSession). * Risks in session handling (session hijacking, fixation). * Cross-Site Request Forgery (CSRF) and methods for prevention (CSRF tokens). * Introduction to Web Application Firewalls (WAF) and their role in web security. * Using OWASP ZAP and Burp Suite for vulnerability scanning and penetration testing. | * Pro JSP by Apress, Chapter 8 (Session Management and Security). * OWASP CSRF Prevention Cheat Sheet. * "The Web Application Hacker's Handbook" by Dafydd Stuttard, Chapter 20 (Testing Tools and WAFs). * OWASP ZAP User Guide (online resource). |
| 10 | **Secure Development Practices in Java**   * Best practices for secure coding in Java (input validation, output encoding, avoiding insecure libraries). * Using Java security features (JAAS - Java Authentication and Authorization Service). | * "Java Security: Writing Secure Code" by Scott Oaks, Chapter 5 (Secure Coding Practices). * OWASP Secure Coding Practices Quick Reference Guide. |
| 11 | **HTTPS, SSL/TLS and Certificate M**  - Secure communication protocols (SSL/TLS) and their use in securing web traffic.  - Setting up HTTPS in web applications (SSL certificates, server configuration).  - Certificate management and trust models. | **-**  Pro JSP by Apress, Chapter 10 (SSL/TLS and Web Security).  - OWASP Transport Layer Protection Cheat Sheet. |
| 12 | **API Security and Secure Communication,**  **Course Review and Final Project**  - RESTful APIs in Java using JAX-RS (Java API for RESTful Web Services).  - Authentication and authorization of APIs (OAuth2, JWT).  - Protecting APIs from attacks (rate limiting, input validation).  - Review of all major web security topics.  - Guidelines for the final project: Developing a secure web application. | **-**  "RESTful Java with JAX-RS" by Bill Burke, Chapter 9 (Security in REST APIs).  - OWASP REST Security Cheat Sheet.  - The Web Application Hacker’s Handbook by Dafydd Stuttard, Chapter 21 (Putting It All Together). |
|  | **Final Project** |  |

**Instructor of the course \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Head of the department \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**