Prague City
Study Centre

In Course Assessment

Module Title: Design Patterns in Java	**Module Leader: Bohus Ziskal**
	Module Code: PRA3016-N
Assignment Title: Food delivery solution	**Deadline Date: 13th January 2026**
	Deadline Time: 23:55
	Submission Method: Online (Blackboard)

Submission Notes:

- Please follow carefully the instructions given on the Assignment Specification
- When an extension has been granted, a fully completed and agreed Extenuating Circumstances form must be submitted

FULL DETAILS OF THE ASSIGNMENT ARE ATTACHED
INCLUDING MARKING & GRADING CRITERIA

Module Assessment Specification:

Module Learning Outcomes:

This assessment specification evaluates the following learning outcomes:

Personal and Transferable Skills

PTS1 - Work efficiently in a team and engage actively in negotiation of goals and methods. (PTS4)

PTS2 - Analyse and critically evaluate sources of information to achieve a comprehensive survey of a given area. (PTS8)

PTS3 - Solve a software development related problem while selecting the appropriate methods, techniques, and tools. (PTS5)

Research, Knowledge and Cognitive Skills

RKC1 - Show proficiency in using a set of advanced programming techniques and skills. (RKC1)

RKC2 - Analyse and select appropriate design patterns and techniques. (RKC2)

Professional Skills

PS1 - Evaluate critically your performance as a team member in software development related projects. (PS1, PS2)

Module Aims:

This module aims to develop and improve abilities and skills in the following aspects:

1. Be able to utilise and judge the significance of design patterns and principles for effective programming in Java and in general in other OO languages.

2. Learn to exploit the power of concurrent programming techniques in Java.

3. Obtain robust knowledge of enterprise-related solutions with Java, including technologies such as Java Beans, JDBC or JAXB.

4. To work efficiently and with minimal supervision in small-teams during thedevelopment of a software application.

Module Assessment Requirements:

Task:

Component 1 (80%) is a small-team (2-3 members) assignment that has two, equally marked, elements. It assesses learning outcomes PTS1, PTS3, RKC1 and RKC2.

Element 1: The task will require students to build a software application in Java based on the relevant design patterns. Students will be assessed for the quality of the software application, its functionality, correct coding techniques and the understanding of the relevant patterns; they will also be assessed for effective participation in a team-work.

Element 2: It requires students to build a software application following the concurrent programming techniques in Java. Students will be assessed for the quality of the application, its functionality, and correct use of concurrent programming techniques.

Rationale

Design patterns are essential in software development because they provide proven, reusable solutions to common design problems, improving code quality and maintainability. In Java, design patterns help developers write flexible, scalable, and easier-to-understand programs by promoting best practices such as loose coupling, encapsulation, and modularity. For example, the Observer pattern is widely used in Java's event-driven programming model, such as in GUI frameworks, to decouple event publishers from subscribers. By applying design patterns, Java developers can reduce development time, minimize errors, and build software that is easier to extend and maintain.

Situation/Scenario

You are expected to demonstrate your ability to apply design principles and design patterns on a real project. You will work in small teams (2-4 students) in an agreed role and actively participate in creating well documented solution that meets the initial requirements.

A startup wants to build a **Food Delivery Application** (similar to UberEats, Zomato, or DoorDash). They want to allow users to order food from restaurants, pay using different methods, and track delivery status in real time.

Since the app will grow in the future, the company wants to use **design patterns** to make the code flexible, maintainable, and scalable.

Functional requirements

F1 – Users can browse different restaurants.
F2 – Each restaurant can offer multiple dishes (Pizza, Burger, Sushi, etc.).
F3 – Users can order dishes from a chosen restaurant.
F4 – The app should be able to offer different types of dishes (Veg, Non-Veg, Vegan, etc.).
F5 – Users can add (order) extra toppings to dishes (Cheese, Extra Sauce).
F6 – Users can pay with **Credit Card, PayPal, or Cash on Delivery**.
F7 – Users should be notified when their order status changes (e.g., *Preparing → Out for Delivery → Delivered*).
F8 – Users can apply different discount strategies (e.g., *Percentage Discount, Flat Discount, No Discount*).

Non-functional requirements
NF1 – the application is initially implemented using command-line interface
NF2 – the system can be easily configured (only one instance of application configuration that stores global settings, e.g., app name, version).
NF3 – the solution provides good support for maintenance and expandability by utilising the design principles

Tasks

- Discuss the requirements in teams, create a list of your solution features and assign roles in team, where each team member will be responsible for creating specific solution functionality.
- Element 1 – address first 5 functional requirements
- Element 2 – address the rest of functional requirements, make sure all non-functional requirements are addressed
- Create the initial (naive) design, which will serve as a base for implementing the desired functionality. The proposed solution architecture will be described using a block scheme and text description, proposed objects and their relationship are presented in a form of UML class diagram.
- Identify and apply suitable design patterns to your initial design in order to make your solution efficient and expandable.
- Describe application of all design patterns using UML class diagrams and in formatted text, justify application of design principles and specific design patterns.
- Implement the suggested design by using appropriate Java programming techniques.
- Use adequate communication tools and some version of control system (VCS) to manage your code, describe your teamwork strategy in your report.

The application does not have to be fully deployed. You can implement only key parts, where design patterns were applied. It should be clear how functional and non-functional requirements are met. There should be summary of team achievements provided and description of project/code management tools used.

Assessment Criteria	Indicative Weighting
Analysis: All main functional blocks (entities) are properly identified and the basic relationships between them are identified. The main concept of the proposed solution is properly defined.	25
Application of design patterns: The design patterns are properly incorporated into the initial design. The usage of each pattern is correctly justified.	35
Implementation: The implementation follows the prepared design and it uses correctly Java programming techniques for the implementation of the suggested design patterns.	25
Coordination: The whole project is well distributed among programmers. The project is also well maintained by utilizing some version control systems (VCS).	15

The total is **100** points accounting for **80%** of the final module grade.

Deliverables and Submission Details

All content, both team work and individual contribution, should be delivered in **the form of technical report in pdf format**.

In the report, all team members and their roles should be clearly identified.

Each team member delivers a unique report. Initial design scheme, UML class diagrams and source code will be the same for all team member reports, the rest of report should be written individually.

A commented source code should be included in the report appendix.

Report expected length - 4000-7000 words (excluding appendix).

The report can be written using any text editor that enables text formatting.

The report is submitted in pdf format through BB LE and it is checked by TurnitIn.

Module Submission: Online

Module Marking Criteria:

Level 6 Grading Criteria

90-100%	Exceptional work. Presentation is logical, error-free and, where appropriate, creative. There is an in-depth understanding of issues/problems and excellent critical/deep engagement with the material and concepts involved. Very skilful interpretation of data. Arguments, ideas and, where appropriate, solutions are presented coherently and fully underpinned by thorough research and reading.
80-89%	Outstanding work with presentation of a very high standard. There is comprehensive understanding of key concepts and knowledge and clear evidence of critical analysis and insight. Accurate interpretation of data with arguments, ideas and solutions presented effectively and based on strong research and reading.
70-79%	Extremely good work with presentation of a high standard. Demonstrates an excellent knowledge base with a clear understanding of the issues and application to practice where appropriate. There is some effective critical and analytical application of relevant research and reading.
60-69%	The work is very good, logically structured and presented to a high standard. Demonstrates a strong knowledge base with a clear understanding of the issues and application to practice where appropriate. There is some critical and analytical application of relevant research.
50-59%	The work is clearly presented and logically structured. It shows evidence of a sound understanding of the topic and addresses major issues. The work contains some discussion and interpretation of relevant perspectives although further development of the arguments presented would be beneficial. There are examples of critical reflection and evidence of application of theory to practice.
40-49%	Adequate presentation. The work displays basic knowledge and understanding of the topic but is largely descriptive. There is an attempt to bring together different ideas and concepts although this would have been strengthened by the inclusion of further key issues. The structure of the work requires attention to its coherence and logical development of content. The link between theory and practice, where appropriate, is somewhat tenuous and its development would enhance the work considerably.
30-39% - Fail	The work is poorly presented and contains numerous errors, inconsistencies and omissions with limited use of source material. The work displays a weak knowledge base and a lack of sufficient understanding of the topic. There is limited evidence of the application of theory to practice where appropriate. It contains many unsupported statements with limited attempts to bring issues together and lacks critical analysis and reflection.
20-29% - Fail	The work is very poorly presented and contains numerous serious errors, inconsistencies and omissions with little use of source material. The work displays a very weak knowledge base and a lack of sufficient understanding of the topic. There is very little evidence of the application of theory to practice where appropriate. It contains many unsupported statements with very little attempt to bring issues together and there is a complete lack of critical analysis and reflection. To obtain a mark of 20% the work must show evidence of a genuine attempt to engage with the assessment requirements and with the subject matter.
0-19% - Fail	The work is extremely poorly structured and presented. It demonstrates no real knowledge or understanding of key concepts and principles. Much material is irrelevant, incorrect, inconsistent or omitted. No evidence of critical analysis and reflection. No effective use of supporting material. No application of theory to practice where appropriate. Not a genuine attempt to engage with the assessment requirements and/or subject matter.

Prague City
Study Centre