LaSalle College
Montréal

Final Evaluation: 40%

Course Identification

Name of program(s) – Code(s):	COMPUTER SCIENCE TECHNOLOGY PROGRAMMING (420.BP)
Course title:	**IOS MOBILE DEVELOPMENT II**
Course number:	420-DM6-AS
Group:	07488
Teacher's name:	Daniel de Rezende Barbosa Carvalho
Duration:	3 hours
Semester:	Winter 2022

Student Identification

Name: _____ Student number : _____

Date : _____ Result: _____

☐ I declare that this is an original work, and that I credited all content sources of which I am not the author (online and printed, images, graphics, films, etc.), in the required quotation and citation style for this work.

Standard of the Evaluated Competencies

Statement of the evaluated competency – 00SY

Collaborate on the design of applications – 00SY

Evaluated elements of the competencies

3. Develop the detailed design.

Statement of the evaluated competency – 00SV

Develop data exchange services - 00SV

Evaluated elements of the competencies

1. Analyze the application development project.
3. Prepare the database.
4. Program the application logic for the service.

Competency: Collaborate on the design of applications – 00SY

General ministerial and institutional performance criteria:

− Demonstration of intellectual curiosity.
− Demonstration of critical thinking.
− Demonstration of analytical thinking.
− Sense of organization

Elements of the competency For 420.BP (2 and 3 only)	**Performance criteria specific to each element**
1. Participate in the development of the functional specifications.	1.1 Accurate analysis of the client's request and requirements 1.2 Accurate analysis of the features of the computer equipment and applications used by the client 1.3 Appropriateness of the recommendations regarding the nature of the requirements 1.4 Appropriateness of the recommendations regarding application development standards, methods and best practices
2. Participate in the overall design of the applications.	2.1 Appropriateness of the recommendations regarding the choice of software architecture 2.2 Sound assessment of the software and hardware components to be used 2.3 Appropriateness of the recommendations regarding security measures to be implemented 2.4 Appropriateness of the recommendations regarding test strategies to be used 2.5 Appropriateness of the recommendations regarding the feasibility of the computing solution
3. Develop the detailed design.	3.1 Modelling of a database aligned with user needs 3.2 Clear identification of the initial data in the database 3.3 Clear description of the application logic and interface to generate or program 3.4 Object-oriented modelling compliant with principles of encapsulation, inheritance, composition and polymorphism 3.5 Proper choice or production of algorithms 3.6 Compliance with nomenclature rules

4. Produce design documents.	4.1 Accurate graphical representation of the different models 4.2 Accurate drafting of unit, integration, functional or acceptance test plans 4.3 Active participation in the design review 4.4 Use of appropriate vocabulary 4.5 Compliance with application development standards, methods and best practices

Competency: Develop data exchange services - 00SV

General ministerial and institutional performance criteria:

– Critical thinking;
– Methodical, analytic and synthetic mind;
– Programming efficiency;
– Autonomy;
– Initiative.

Elements of the competency For 420.BP (4,5,6,7 and 8)	**Performance criteria specific to each element**
1. Analyze the application development project.	1.1 Accurate analysis of design documents 1.2 Proper identification of the tasks to be carried out
2. Prepare the computer development environment.	2.1 Proper installation of the development platform and development database management system 2.2 Proper installation of software and libraries 2.3 Appropriate configuration of the version control system 2.4 Proper importing of the source code
3. Prepare the database.	3.1 Suitable creation or adaptation of the database 3.2 Proper insertion of initial or test data 3.3 Compliance with the data model
4. Program the application logic for the service.	4.1 Proper programming or integration of authentication, authorization or secure connection establishment mechanisms 4.2 Proper programming of the reception of input data 4.3 Appropriate choice of clauses, operators, commands or parameters in database queries 4.4 Correct handling of database data 4.5 Proper programming of the response of output data 4.6 Precise application of secure programming techniques

	4.7 Compliance with communication protocols and data exchange formats
5. Program a test application using the service.	5.1 Precise retrieval of the service interface 5.2 Appropriate use of the service 5.3 Proper conversion of the data provided by the service into run data using a test application
6. Control the quality of the service.	6.1 Precise application of test plans 6.2 Thorough reviews of code and security 6.3 Relevance of the corrective actions 6.4 Compliance with issue tracking and version control procedures 6.5 Compliance with design documents
7. Participate in the deployment of the service.	7.1 Proper application of the procedure for migrating the service onto the server 7.2 Precise application of security measures
8. Produce the documentation.	8.1 Proper identification of the information to be written up 8.2 Clear record of the work carried out

Instructions

- Your exam must be submitted by uploading your project via Omnivox. Deadlines are shared on Omnivox in the assignment box and must be respected.

- It is the teacher's responsibility to identify language errors. If such errors are found, teachers may deduct up to 5% of the final grade (IPEL – Article 5.7).

- Plagiarism attempts at plagiarism or complicity in plagiarism during a summative evaluation results in a mark of zero (0). In the case of recidivism, in the same course or in another course, the student will be given a grade of '0' for the course in question. (IPEL – Article 5.16).

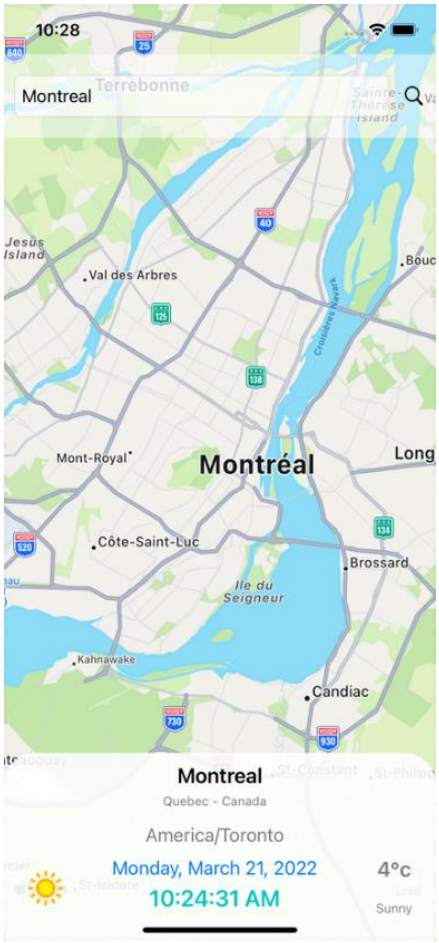
Mark Breakdown

This evaluation is on 100 points, distributed as follows:

Question 1 – Issues report	70 points
Question 2 – Data persistance	30 points
TOTAL:	**100 POINTS**

Case study

The Company ABCD Technology/Montreal is developing an IOS mobile application for an American Company based in San Francisco. This app will be used globally by their executives to show a city's current time based on the time zone and the location's current weather. At the same time, the company wants to create a log with basic information about the requests, saving this **data locally on the user's device**. You should use **Apple's Core Data framework** for persisting data.



IMPORTANT

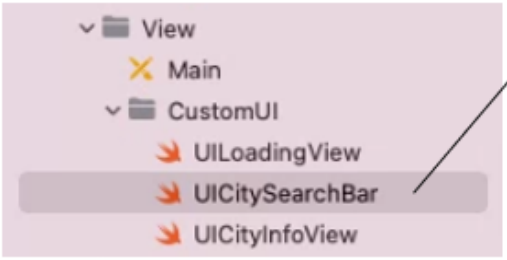
This project involves subjects taught in class about customized user interfaces, API calls, protocol definitions and data persistence. Remember that the user interface should be defined programmatically.

The use of storyboard to manually place the views is not allowed.

QUESTION # 1 **ISSUE REPORT**

Your first task is to follow the issue report, fixing bugs and improving the code as indicated.

□ Issue #1	10 pts	**UICitySearchBarView** class name wrongly defined



UICitySearchBar customized view name is incorrect. The specified name is **UICitySearchBarView**. You have to refact the code, changing the **file**, **class** and **protocol** names for the correct one.

□ **I**ssue #2	10 pts	Incorrect text for place holder

Placeholder is incorrect. The correct text to be presented is "**enter the city name to fetch the time zone**"

□ Issue #3	10 pts	UICitySearchBarView delegate is not working

The protocol function **citySearchBar**(cityToSearch …) from **UICitySearchBarViewDelegate** is not implemented inside **ViewController**. You have to implement it, calling the API with the given city name. Remember that the API should only be called if the city name has a minimum of two characters size. Otherwise, an error message should be presented informing this minimum requirement.

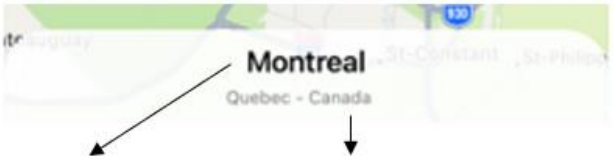
□ Issue #4	10 pts	**MapView** is not showing the desired location

MapView is not showing the location after fetching the city information.

Analyze the fields returned by the API and set the mapView region to show the returned longitude and latitude.

□ Issue #5	10 pts	Labels **lblCity** and **lblRegion** style and constraints wrongly defined on **UICityInfoView**

lblCity and **lblRegion** are not following the designer requirements.



lblCity
Font style: **bold**
Font size: **20**
Text alignment: **Center**

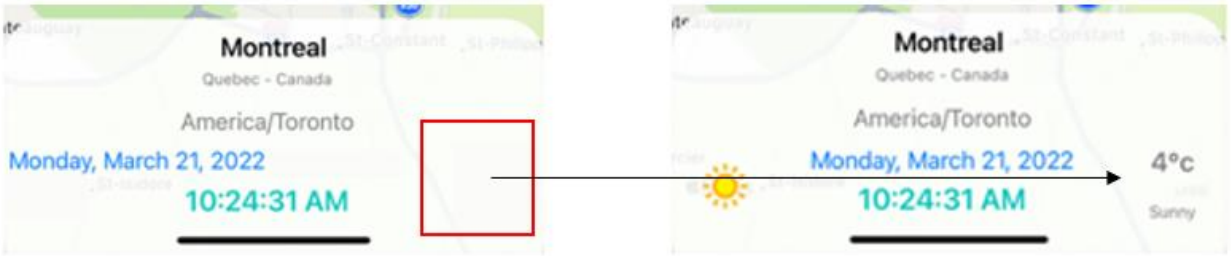
lblRegion
Font style: **Regular**
Font size: default
Text alignment: **Center**

Apply proper constraints for both labels

□ Issue #6	10 pts	Label **lblDate** with incorrect constraints

Label **lblDate** constraints are incorrect. This field was designed to be centered.

□ Issue #7	10 pts	Temperature value and condition text missing



Labels **lblTempC** and **lblCondition** are missing. Those informations come from the API.

lblTempC
Font style: **bold**
Font size: **20**
Text color: **gray**
The **topAnchor** constraint is the same applied for **lblDate**

lblCondition
Font style: **regular**
Font size: **12**
Text color: **gray**
Number of lines: **2**

QUESTION #2 **DATA PERSISTENCE**

PART #1

The company wants to access the history of all users consultations persisted **locally on the device**, using Apple's **Core Data** framework.

You have to design data Model with the mandatory fields and implement the code to insert this information every time a city is fetched.

Mandatory fields:

City,

Device date and time,

Region,

Time zone id,

Temperature in Celsius.

Important: Respect the MVC pattern, creating the classes and functions inside the right group/folder.

Correction Grid

Elements of the competency:
- 00SY: 3. Develop the detailed design

Performance criterion:

3.1 Modelling of a database aligned with user needs
3.2 Clear identification of the initial data in the database
3.3 Clear description of the application logic and interface to generate or program
3.4 Object-oriented modelling compliant with principles of encapsulation, inheritance, composition and polymorphism
3.5 Proper choice or production of algorithms
3.6 Compliance with nomenclature rules

Elements of the competency:
- 00SV: 1. Analyze the application development project.

Performance criterion:

1.1 Accurate analysis of design documents
1.2 Proper identification of tasks to be carried out.

Elements of the competency:
- 00SV: 3. Prepare the database.

Performance criterion:

3.1 Suitable creation or adaptation of the database.
3.2 Proper insertion of initial or test data.
3.3 Compliance with the data model.

Elements of the competency:
- 00SV: 4. Program the application logic.

Performance criterion:

4.1 Proper programming or integration of authentication, authorization or secure connection establishment mechanisms.
4.2 Proper programming of the reception of input data.
4.3 Appropriate choice of clauses, operators, commands or parameters in database queries
4.4 Correct handling of database data.
4.5 Proper programming of the response of output data.
4.6 Precise application of secure programming techniques.
4.7 Compliance with communication protocols and data exchange formats.

Criterion-elements:

User interface programmatically built. Use of storyboard to place views is not allowed.

Proper UI design following given instructions, prototype images (object colors, styles, sizes, margins, alignments).

Well-defined messages and instructions for every user interaction, when applied.

Correct use of constraints, adding user interface proper responsiveness.

Use of variables, classes, UI objects and methods names following patters.

Data model following given instructions. Correct implementation of methods to insert data into cloud or local services.

Correct implementation of success and fail handlers/closures for API calls. Well-defined struct with the required fields decoded from the API JSON response.

Correct use of protocols for communication between views/controllers.

Clear and readable code, applying comments **only** for business rules or relevant information (not for explaining the syntax or logic applied).

No syntax or compilation errors (application running). No logic errors.

Question 1	**Highly satisfactory**	**Satisfactory**	**Unsatisfactory**	**Highly unsatisfactory**	**Total**
Issue #1	10 Completely correct	6 ..< 10 Almost correct	2 ..< 6 Incomplete	< 2 Completely wrong	_____ **/ 10**
Issue #2	10 Completely correct	6 ..< 10 Almost correct	2 ..< 6 Incomplete	< 2 Completely wrong	_____ **/ 10**
Issue #3	10 Completely correct	6 ..< 10 Almost correct	2 ..< 6 Incomplete	< 2 Completely wrong	_____ **/ 10**
Issue #4	10 Completely correct	6 ..< 10 Almost correct	2 ..< 6 Incomplete	< 2 Completely wrong	_____ **/ 10**
Issue #5	10 Completely correct	6 ..< 10 Almost correct	2 ..< 6 Incomplete	< 2 Completely wrong	_____ **/ 10**
Issue #6	10 Completely correct	6 ..< 10 Almost correct	2 ..< 6 Incomplete	< 2 Completely wrong	_____ **/ 10**
Issue #7	10 Completely correct	6 ..< 10 Almost correct	2 ..< 6 Incomplete	< 2 Completely wrong	_____ **/ 10**
				TOTAL	_____ **/ 70**

Question 2	**Highly satisfactory**	**Satisfactory**	**Unsatisfactory**	**Highly unsatisfactory**	**Total**
Part #1	30 Completely correct	18 ..< 30 Almost correct	6 ..< 18 Incomplete	< 6 Completely wrong	**/ 30**
				TOTAL	_____ **/ 30**

Correction Grid for Language

Clear communication	Clear communication most of the time	Vague communication	Unclear communication
- 0	- 0.5	- 1.5	- 2
(Word Choice) Use of precise and rich vocabulary	(Word Choice) Use of precise Vocabulary	(Word Choice) Use of imprecise Vocabulary	(Word Choice) Use of inappropriate vocabulary
- 0	- 0.5	- 1.5	- 2
(Format/Type of work) Respect of norms	(Format/Type of work) Respect of most of the norms	(Format/Type of work) Non-respect of the norms	(Format/Type of work) Inappropriate in relation to the required norms
- 0	- 0.5	- 1.5	- 2
(Linguistic Code) (≤2 mistakes/page)	(Linguistic Code) (3-7 mistakes/page)	(Linguistic Code) (8-10 mistakes/page)	(Linguistic Code) (>10 mistakes/page)
- 0	- 0.5 … - 2.5	- 2.5 … - 3.5	- 4