

## Architecture and mobile application development featuring Sailfish OS

### Course Administrative Details

Course Title	Architecture and mobile application development featuring Sailfish OS		
Instructor(s)	Alexander Borodin	Instructor's e-mail	alexander.borodin@fruct.org
Course #	433	Course Type	Elective
Faculty	Computer Science and Engineering	Major	Computer science
Academic year	2016-2017	Semester Offered	Spring
No. of Credits	6	Total workload on average	30 hrs. per week inc. 10 hrs. of self-study
Lecture Hours	10 per week	Lab Hours	10 per week
Language	English	Frequency	3 times weekly modules per semester
Target Audience	Bachelors	Anticipated Enrollment	25 students
Studying year	4		
Grading Mode	A, B, C, D	Keywords	Sailfish OS, Mobile OS, App development

### Course Outline

This is a general course in mobile development with an emphasis on Sailfish OS. During the course, students will learn the fundamental principles and techniques that can be applied to design and develop mobile applications, and will cover architectures of GUI applications, the usage of mobile sensors API. Important and universal skills of application development using Qt/QML will be gained. The course contains a large number of practical examples, tasks and the necessary practical minimum to start developing apps on Sailfish OS.

### Course Delivery

The course will be given over 15 days. Individual assignments include a series of materials for self-study, one large group assignment for the time between first and second modules, 8 group tasks that will be given for every day during first and second modules and the course project task that will be given in the end of second module. The main assignment for the third module is work on the course project task using help of the course instructor. In addition, the course instructor will be available twice per each module for individual consulting of the students, plus between modules there will be two time slots of 1 academic hour per week that students can reserve for individual consulting over Skype.

Course structure: 3 modules, 1 week each. Each module contains 10 hours of lectures + 10 hours practical labs + assignments and time for consulting.

### Prerequisite courses

It is recommended to pass basic courses: "Introduction to programming", "Data structures and algorithms", "Operating systems", "Programming paradigms" to be familiar with general programming terms, methods and tools.

### Required background knowledge

Solid knowledge of imperative and object-oriented programming concepts (including structures and classes, encapsulation and data hiding, methods, member access control, inheritance, polymorphism) and good programming skills are assumed. Students should also have a working knowledge of data structures, algorithms and operating systems.

**Course structure**

Module 1. Fundamentals of mobile apps development

Date	Topic	Assignments
30.01.2017	Introduction to mobile programming featuring Sailfish OS	RQ, IA
31.01.2017	QML language basics on the examples of Sailfish apps	RQ, IA
01.02.2017	Constructing the fluid mobile UIs with QtQuick and Silica controls	RQ, IA
02.02.2017	Defining UI navigation in multi-page mobile apps	RQ, IA
03.02.2017	Decomposing QML code into components and integrating Javascript	RQ, GA

Module 2. Data processing and interactivity

Date	Topic	Assignments
27.03.2017	Separating data from visualization with models and views	RQ, GA
28.03.2017	Using LocalStorage API and integrating SQLite with QML apps	RQ, GA
29.03.2017	Connectivity management and gathering data from the Web	RQ, GA
30.03.2017	Canvas-based and OpenGL ES 2 graphics for interactive apps and games	RQ, GA
31.03.2017	Multimedia capture and playback	RQ, IA

Module 3. Advanced mobile programming

Date	Topic	Assignments
24.04.2017	Using the services of Sailfish mobile platform	IA
25.04.2017	Processing the data from on-board sensors, location-based services and augmented reality	IA
26.04.2017	Extending QML apps with C++	IA
27.04.2017	Using the tools for quality assurance and best practices	IA
28.04.2017	Packaging and distributing the apps through the store and public repositories	IA

**Textbook(s)**

Qt5 Cadaques by Jürgen Bocklage-Ryannel and Johan Thelin <https://qmlbook.github.io/>

Mobile Usability by Jakob Nielsen & Raluca Budiu, 2012 ISBN-13: 978-0321884480

Professional Mobile Application Development by Jeff McWherter, Scott Gowell, 2012 ISBN 978-1-1182-0390-3

**Reference Materials**

- Sailfish OS website – <https://sailfishos.org>
- QML Applications – <http://doc.qt.io/qt-5/qmlapplications.html>
- Creating applications with Sailfish Silica – <https://sailfishos.org/develop/docs/silica/>
- Sailfish Silica Reference – <https://sailfishos.org/develop/docs/silica/sailfish-silica-all.html/>
- Sailfish Icon Reference – <https://sailfishos.org/develop/docs/jolla-ambient/>
- Qt Documentation <http://doc.qt.io/>

**Computer Resources**

Students should have PCs (better laptops, to use the same environment in the class and at home) supporting hardware virtualization with installed Sailfish OS SDK and VirtualBox.

### Laboratory Exercises

1. Setting up and using Sailfish SDK
2. Using the IDE, command-line tools and toolchains for preparing and deployment Sailfish OS template
3. Creating QML using interfaces
4. Designing and implementing UI navigation
5. Using mobile UI controls
6. Manipulating the data with ListModels
7. Managing the data sources and consumers
8. Integrating mobile apps with the web APIs
9. Designing and implementing the gaming app
10. Playing the sounds and music on mobile devices
11. Working with D-BUS and platform services
12. Processing the accelerometer data
13. Binding the QML app with C++ code
14. QML profiling and optimizing the code
15. Publishing the application

### Laboratory Resources

- Sailfish OS driven mobile devices supporting accelerometer, Bluetooth, geolocation.
- PCs with Sailfish OS SDK, regular Qt SDK and VirtualBox.

### Grading criteria

Assignments (50%), Course project (50%)

### Late Submission Policy

This policy will be strictly applied in this course. If a personal emergency should arise that affects your ability to turn in an assignment in a timely fashion, you must contact the course instructor BEFORE the deadline to get a "Special Late Submission Approval" from the course instructor. Without the "Special Late Submission Approval" submissions will be still accepted up to a week late, but with a 50% penalty. No "Special Late Submission Approval" will be granted after the deadline. All late submissions should be sent by email directly to the instructors.

### Cooperation Policy and Quotations

We encourage vigorous discussion and cooperation in this class. You should feel free to discuss any aspects of the class with any classmates. However, we insist that any written material that is not specifically designated as a Team Deliverable be done by you alone. This includes answers to reading questions, individual reports associated with assignments, and labs. We also insist that if you include verbatim text from any source, you clearly indicate it using standard conventions of quotation or indentation and a note to indicate the source.