

# **Project descriptions**

A summary of development projects, both in college and private

Knut Lucas Andersen

January 27, 2017

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Visual Studio: C#</b>	<b>4</b>
2.1	Real-estate registration . . . . .	4
2.2	Bachelor Thesis: CleanMyFolder . . . . .	4
2.3	Computer Graphics . . . . .	4
2.3.1	Graded Assignment: Car game . . . . .	4
2.4	Model Train registration . . . . .	4
2.5	Multipurpose editor . . . . .	5
<b>3</b>	<b>Visual Studio: C++</b>	<b>6</b>
3.1	C++ course . . . . .	6
3.1.1	Graded assignment: recipeProgram . . . . .	6
3.2	Machine Learning library . . . . .	6
3.3	Computational Forensics . . . . .	6
<b>4</b>	<b>Java and Android</b>	<b>7</b>
4.1	Animal Registration . . . . .	7
4.2	Android course . . . . .	7
4.2.1	TracknHide . . . . .	8
4.3	TrackMyTeacher . . . . .	8
4.4	BeregnSnitt . . . . .	9
4.5	Auto-reply for Android . . . . .	9
<b>5</b>	<b>Python</b>	<b>10</b>
5.1	Global Software Development . . . . .	10
5.2	Advanced Project Work . . . . .	10
5.3	Master thesis: Predicting coding question quality . . . . .	10
<b>6</b>	<b>Web development</b>	<b>11</b>
6.1	Alumni Community . . . . .	11
6.2	Flea Market . . . . .	12
6.3	Applied Computer Science Project . . . . .	13
6.4	Integration project . . . . .	14
<b>7</b>	<b>Others</b>	<b>15</b>
7.1	Teacher-Student questionnaire system . . . . .	15
7.2	System Developing Project . . . . .	15
7.3	Flash Programming . . . . .	15
7.3.1	Space Invaders . . . . .	16
7.3.2	Tower of Knowlegde . . . . .	16
7.4	3D Modelling . . . . .	16

# 1 Introduction

This document lists all the development projects I have been a part of, starting from my Bachelors degree until my completed Masters degree in Applied Computer Science. Each section is dedicated to a given programming language, the exceptions are the two last sections, Section ?? and ??. Projects uploaded to GitHub will also contain a link to its given repository, but group projects is not listed. For all projects, there will be a listing for project type (Assignment or Private), IDE, Editor, Tools and Programming language(s) that were used. Development projects from courses will also include the College/University, the Level (Bachelor or Master) and the Course it belongs to.

Repository: <https://github.com/klAndersen?tab=repositories>

## 2 Visual Studio: C#

### 2.1 Real-estate registration

- Level: Bachelor
- Project type: Assignment
- College/University: Buskerud University College
- Course: INF313 Management of Databases
- IDE, Editor, Tools:
- Oracle-Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/Databases/INF%20315%20-%20Oracle/Oblig1>
- C#-Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/C%23/Oblig%20Inf315>

### 2.2 Bachelor Thesis: CleanMyFolder

- Level: Bachelor
- College/University: Buskerud University College
- Course: INF350 Bachelor Thesis
- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/C%23/Bachelor%20thesis>

### 2.3 ITE1605 Computer Graphics

- Level: Bachelor
- College/University: Narvik University College
- Programming language(s): C#/XNA
- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/XNA/Datamaskingrafikk>

#### 2.3.1 Graded Assignment: Car game

Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/XNA/Datamaskingrafikk/Karaktergivende%20oppgave>

### 2.4 Model Train registration

Project type: Private

Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/C%23/ModellTog>

## 2.5 Multipurpose editor

- Project type: Private
- IDE, Editor, Tools:
- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/C%23/MultiPurpose>

## 3 Visual Studio: C++

### 3.1 ITE1546 Programming C++

- Level: Bachelor
- Project type: Assignments
- College/University: Narvik University College
- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/C%2B%2B/ITE1546%20Programmering%20i%20C%2B%2B>

#### 3.1.1 Graded assignment: recipeProgram

Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/C%2B%2B/ITE1546%20Programmering%20i%20C%2B%2B/Karaktergivende>

### 3.2 Machine Learning library

- Level: Master
- Project type: Developed for re-usability in assignments
- College/University: Gjøvik University College
- Course: IMT4612 Machine Learning and Pattern Recognition 1
- Repository: <https://github.com/klAndersen/Machine-Learning/tree/master/MachineLearning>

### 3.3 IMT4641 Computational Forensics

- Level: Master
- Project type: Assignment
- College/University: Gjøvik University College
- IDE, Editor, Tools: Visual Studio, SQLite
- Repository: <https://github.com/klAndersen/IMT4641-Computational-Forensics>

## 4 Java and Android

### 4.1 Animal Registration

- Level: Bachelor
- Project type: Assignment
- College/University: Buskerud University College
- Course: INF240 and INF244 Object Oriented Programming
- IDE, Editor, Tools: Eclipse
- Programming language(s): Java
- Repository: [https://github.com/klAndersen/Bachelor-development-projects/tree/master/Java/Oblig2\\_Inf240\\_KnutLucasAndersen](https://github.com/klAndersen/Bachelor-development-projects/tree/master/Java/Oblig2_Inf240_KnutLucasAndersen)
- Repository: [https://github.com/klAndersen/Bachelor-development-projects/tree/master/Java/Oblig\\_Inf244\\_KnutLucasAndersen](https://github.com/klAndersen/Bachelor-development-projects/tree/master/Java/Oblig_Inf244_KnutLucasAndersen)

In the two Java courses in my Bachelor, we had the same assignment, but with different requirements. The main focus was to register two different animals (hares and lynx). Common for both was the registration of their gender (male/female), length (Double), weight (Double), time of capture (day, month, year) and location (String). The ID was to be incremented, and start with the initial for the given animal (e.g. L1, L2, ..., H1, H2, ...). For the hares, colour (String) and type (Char) was to be registered. For the lynx, the length of their ears (Double).

If an animal was later re-captured, their data should be updated instead of re-registered. The program had the possibility to search for animals based on ID, show amounts of re-captures (based on the entered year), show amount of different hare captures (based on type), and an unsorted report of all first-time captures.

The first course was introductory level, so input/output was shown in command line, and data was stored in a text file. In the second course, the user was presented with a GUI that we had to code ourselves (not use a Designer), and the data was stored in a MySQL Database (using JDBC as database driver). Singleton was used to ensure that there was only one object maintaining the database connection. The main menu used JFrame, and the child windows used JDialog.

### 4.2 ITE1621 Applications for mobile and web

- Level: Bachelor
- Project type: Assignment
- College/University: Narvik University College
- IDE, Editor, Tools: Eclipse
- Programming language(s): Java, Android, XML

- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/Android/ITE1621%20Applikasjoner%20for%20mobil%20og%20web>

#### 4.2.1 TracknHide

This project was part of the course grade, and consisted of developing an application that could show the different users location and route on a map. The project had two parts, an Android application and a Java web server.

The Android application used Google Cloud Messaging (GCM) to send and receive data, and Google Maps (API v1) to present a map. By using MapOverlay, users could also show the address when clicking a location on the map, and switch between Street – and SatelliteView. On the map, the route and current position of logged in users with shared position was shown. Users could also choose to store their own or others route, which could later then be re-drawn on the map. The position data was retrieved by using the devices GPS, but the amount of position data sent was set by the user (amount of time passed or distance moved).

The Java Web server was created using HTML, Java Servlets and JSP, running on Apache TomCat v7.0. It stored the user related information in a MySQL Database, and kept track over all the currently logged in users. The web pages was mainly targeted at an administrator, to give them the ability to view connected users, and disconnect if needed. The server was also the connection point for the users to share/receive position data and notifications via GCM. Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/Android/ITE1621%20Applikasjoner%20for%20mobil%20og%20web/Karaktergivende%20oppgave>

### 4.3 TrackMyTeacher

- Level: Master
- Project type: Assignment
- College/University: NTNU
- Course: IMT5401 Advanced Course in Mobile Technology
- IDE, Editor, Tools: Android Studio, MySQL Workbench
- Programming language(s): Android, XML
- Repository: <https://github.com/klAndersen/IMT5401-Mobile-Research>

In this course I tried to develop an indoor tracking prototype app for Android. The purpose was to be able to locate teachers and professors on campus, by measuring the WiFi-signals at their current location (if they had enabled/allowed tracking of their current location).

The server side of the application was developed using PHP (because the college server only ran PHP), which mainly handled data transfer between the Android app and the MySQL database. When data from the database was



needed, the server retrieved the data, added it to an array as JSON and printed it as HTML. This HTML was then converted to a JSONArray on the mobile device (as I stated in my report, it would probably have been much easier if one could have used Java Servlets, since one could overwrite the HTTP GET/POST methods).

To get test data, I went to different buildings, rooms and floors spread out on campus (20 rooms, signals measured in all 4 corners plus the middle of the room). What I found when looking at the gathered data was that the measurement for the WiFi-signal was set too low (value=5). Therefore this prototype was not finished, as I did not have the time to re-measure all the rooms to get more accurate signal data.

#### 4.4 BeregnSnitt

- Project type: Private
- IDE, Editor, Tools: Eclipse
- Programming language(s): Java
- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/Java/beregnSnitt>

When nearing the end of my Bachelor degree, I wanted to know what my average grade was, to see what colleges and universities I could apply to in regards to taking a Master degree. This led to the development of “BeregnSnitt”. BeregnSnitt was a small, private Java project, which calculates an average score based on the grades the user has.

The user is presented with a GUI (Jframe and Jdialog), where s/he can enter the amount of A to E’s achieved, the course points and the total points achieved (e.g. Bachelor = 180). I also added the possibility to add additional course grades, in case some courses varied (e.g. if the standard was 10, but one course had 3 points).

A Norwegian Java application I developed that calculates the average score of the grades achieved within a college. The main reason I created it was to see what colleges and universities I could apply to based on my current grade average. The user can enter the amount of A to E’s achieved, the course points and the total points achieved (e.g. Bachelor = 180). One could also add additional grades (e.g. if some courses had different scoring). Then the user had to enter the grade and the course points.

#### 4.5 Auto-reply for Android

- Project type: Private
- IDE, Editor, Tools: Eclipse, Android Studio
- Programming language(s): Java, Android, XML

## 5 Python

### 5.1 IMT4112 Global Software Development

- Level: Master
- Project type: Group, Assignment
- College/University: NTNU
- IDE, Editor, Tools: PyCharm, Arch Linux (OS)
- Programming language(s): Python

### 5.2 IMT5251 Advanced Project Work

- Level: Master
- College/University: NTNU
- IDE, Editor, Tools: PyCharm, Arch Linux (OS)
- Programming language(s): Python
- Repository: [https://github.com/klAndersen/IMT5251\\_AdvProjWork](https://github.com/klAndersen/IMT5251_AdvProjWork)

### 5.3 Master thesis: Predicting coding question quality

- Level: Master
- College/University: NTNU
- Course: IMT4904 Master Thesis
- IDE, Editor, Tools: PyCharm, Arch Linux (OS)
- Programming language(s): Python 2.7, Python 3.4 (submitted version)
- Repository: [https://github.com/klAndersen/IMT4904\\_MasterThesis\\_Code](https://github.com/klAndersen/IMT4904_MasterThesis_Code)

## 6 Web development

### 6.1 Alumni Community

- Level: Bachelor
- Project type: Assignment
- College/University: Buskerud University College
- Course:
- IDE, Editor, Tools:
- Programming language(s):
- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/Web%20Development/Inf%20268%20Utviklingsprosjekt>

This was a course that focused mainly on development for Web. In this course, everything was developed from scratch, without use of external frameworks or libraries. The goal was to develop an alumni community web page for the college. A requirement was that the delivered solution should include HTML, CSS3, JavaScript and PHP. For storing data, MySQL was selected as database. To have a local server running, Apache XAMPP was used. It was also a requirement that the pages should look as identical as possible regardless of the web browser that was used (I compared Chrome, Internet Explorer, Opera and Firefox).

The course was split into two semesters, and the development project into five parts:

1. Web page design/layout and policy
2. Access Control
3. Functionality and database model
4. Posts and user administration
5. Completion

The users were divided into the following user groups; registered, moderator and administrator. Administrators was able to appoint new moderators and edit users, with the same rights as moderators.

All registered users should be able to...

- ...register, login, update their profile and be able to change/request a new password (if forgotten).
- ...upload a profile picture, or select one of the existing ones that had previously been uploaded to the alumni page.
- ...join or leave networks created within the alumni community. Users should be able to message each other, and also see a notification when new messages has arrived.

- ... see a list of the other registered users.

Moderators should be able to...

- ... edit users profile (e.g. if inappropriate data was entered on their profile).
- ... notify users they had breached the guidelines or block out/quarantine users (users should not be deleted).
- ... create/edit/delete existing topic fields (e.g. events, area of expertise, etc).

In my project I mostly used PHP and HTML. Every page was created as a .php, as this allowed me to update the page layout in just one file, instead of having to update all the pages. The layout was then included at the location where it was to be used. It also simplified the use of function calls and checks to see if a user was logged in, and had the proper rights to view a given page. Database access was also handled via php, and JavaScript was not largely used (mostly to verify user input).

The following is a short summary of what the delivered version included. For new users, they had a registration form, that gave feedback both with JavaScript and PHP if an error occurred. If the user continued, even though one or more fields contained errors, the given field(s) would have a text marked in red next to it, explaining what was wrong. Users also got their own profile, which allowed them to update their information, alongside changing their password (users that weren't logged in could request a new password if they forgot it). Moderators also had the ability to edit user profiles, e.g. if any added something in-appropriate.

Users could also send messages to each other, and to ensure that the users would see if they had received a new message, the link to the messages in the navigation bar was updated with a number, indicating how many new, unread messages they had received. Users could message each other via their profile, or by searching for them from the member list (which listed all, or those matching the search criteria; name or e-mail). Users who broke the page policy could be put in quarantine. This rejected the users attempt to login, and giving them a message that they were quarantined. An information message could also be set, that would explain to the given user why they were quarantined.

Moderators could register campus, area of expertise and networks (e.g. degree, courses, etc). Moderators could also create events. All of the aforementioned could be updated or deleted. For events, only those active were displayed (but administrators and moderators could see all on the creation page). Events that were out of date was not showed, and this was also true for future events (based on start date for the given event).

## 6.2 Flea Market

- Level: Bachelor
- Project type: Assignment

- College/University: Buskerud University College
- Course:
- IDE, Editor, Tools:
- Programming language(s):
- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/Web%20Development/Inf%20329%20XML%20-%20blig.%20oppg>

A school band arranged yearly a flea market, to increase their income. This was a process lasting for months, starting with organization of the sales, to finally selling the various sales items. The flea market consists of several departements, several people are involved and also a lot of money. Income overview and control is a necessity to be able to plan next years flea market. Up until now, they used Excel sheets to keep an economical overview (participants sorted by departements). Every participant gets a bag of change, and can deliver income several times throughout the day. The participants on a Saturday is not necessarily the same as those on a Sunday (sales were only on Saturdays and Sundays). A daily overview over departement sales is registered, and a control is done to check the money delivered against the registered income.

The goal was to re-write this web-application to use XML as data storage, as the currently database in use (Access) gave compability issues depending on the version used. The assignment contained attachments showing screenshots of the Web-page, the Excel file and how the Access database looked. A requirement was that the file containing the database data should be in XML, and that this file should be controlled by using XSD (since the XML file could be altered manually). It was also a requirement to create at least four different web-pages to show how they could use the created XML-files to present and store new data.

### 6.3 IMT4003 Applied Computer Science Project

- Level: Master
- Project type: Group, Assignment
- College/University: NTNU
- IDE, Editor, Tools:
- Programming language(s):

In this course, our group worked on developing an A/B Test system. The following is a short project description, taken from our group report:

”The product is an Internet application, where the goal is to track user interaction. The developer/administrator is able to see the user interaction real time, and the interaction will also be stored in a database, to be shown for the developer/administrator later. This functionality is relevant when a person or a company wants to know the best possible user interface for a website. With this Internet

application, you can present an interface for a user group, and then you can compare more interfaces with each other, to decide which one the user interacts with most efficient.”

Various frameworks were used in this project, but I those I was directly involved with was the Java Servlets and MySQL. In the development part, I had the responsibility for the database model, the database and the database operations. Since we used Model-View-Controller (MVC), some of the programming I did was related to the interfaces and connections between the GUI, the Controller and the Database.

## 6.4 IMT4004 Integration Project

- Level: Master
- Project type: Assignment
- College/University: NTNU
- IDE, Editor, Tools:
- Programming language(s):
- Repository: <https://github.com/klAndersen/IMT4004-Integration-Project>

The original plan for this project was to create a 3D-simulation game that incorporated Darwins theory of evolution. The goal was to try helping students taking Machine Learning courses to learn an algorithm called "Genetic Algorithms". However, due to time restraints and complexity, this was changed and reduced to an unfinished web-page version (aimed at teaching Genetic Programming).

This version consisted of pages for students and teachers, where teachers could create new assignments, check student participation and adjust student scores based on their answers/progress. The students had access to creating and editing their profile, see game progress and score, and they could play a given game an unlimited number of times (but attempts and answers were logged in the database). After submission, the given game would no longer be playable.

The only game-element that was implemented was a LISP (reverse polish notation) game. The student was presented with an equation (e.g.  $1 + (2 * 3) + (10 * 10) + 30$ ) and asked to write the LISP version of this. In addition, the student could see their answer represented as a tree-diagram (achieved by converting the input to a JSON-object, and passing it to the d3.js library).

The project was developed by using HTML5 and PHP, and included a data-visualization library called d3.js. D3.js was only used for creating the LISP tree, which was optional when the student were checking their answer. The data was stored in a MySQL database.

## 7 Others

Suggestions for sections:

- databases
- udk/unreal
- INF340 Information Systems in Business
- ITE1607 Game Design
- IMT4006 Intro to research on Web technologies
- IMT4032 Usability and Human Factors in Interaction Design
- IMT4072 Cross-media color reproduction
- IMT4122 Software Security Trends
- IMT4612 Machine Learning and Pattern Recognition 1: assignments
- anything else...?

### 7.1 Teacher-Student questionnaire system

- Level: Bachelor
- Project type: Group, Assignment
- College/University: Buskerud University College
- Course: INF116 Object Oriented Programming
- IDE, Editor, Tools: Visual Studio, MySQL Workbench
- Programming language(s): Visual Studio/.NET Visual Basic (VB)

### 7.2 INF165 System Developing Project

- Level: Bachelor
- Project type: Group, Assignment
- College/University: Buskerud University College
- IDE, Editor, Tools: Microsoft Access, MySQL Workbench

### 7.3 INF330 Flash Programming

- Level: Bachelor
- Project type: Group, Assignment
- College/University: Buskerud University College
- IDE, Editor, Tools: Adobe Flash CS 4, FlashDevelop
- Programming language(s): Flash, ActionScript 3.0

### 7.3.1 Space Invaders

-details-

### 7.3.2 Tower of Knowledge

-details-

## 7.4 ITE1606 3D-Modelling

- Level: Bachelor
- Project type: Assignment
- College/University: Narvik University College
- IDE, Editor, Tools: 3DS Max
- Repository: <https://github.com/klAndersen/Bachelor-development-projects/tree/master/3D%20Modeling/ITE1606%203Dmodelling>