**Your Name** <your-ju-id@student.ju.se>

A Project Work in *Web Development Fundamentals*

Jönköping University 2019

Your title here...

*In this template, all italic text should be removed and replaced with your own text (which should not be italic); the italic text is just a placeholder letting you know what to write there.*

*On the cover page, change to your own project name, your own name and your own JU email address.*

*You have a lot of freedom when it comes to writing this report. You do not have to use any part of this template, but the report you write should in the end somehow (and in a good way) provide the same information as indicated in this template. Most students trying to do it in their own way usually fail, so if you try that, be sure to know what you are doing!*

*Figures are great, use a lot of them because it's easier to readers to look at figures than reading text, but be sure to give each one a figure number and a caption, and refer to them from the main text using their figure numbers.*

*This page should be removed.*

Table of Contents

[Introduction 3](#_Toc12111050)

[Architecture 4](#_Toc12111051)

[Database 5](#_Toc12111052)

[Web Application 6](#_Toc12111053)

[Graphical User Interface 6](#_Toc12111054)

# Introduction

*Introduce your project work here. Remember that a project is about solving a problem, so be sure to explain what problem you're trying to solve and what your solution to that problem is. Write text that* ***indirectly*** *answers questions like:*

* *Why does the project exist?*
* *What is the project about?*
* *Who are involved in the project?*
* *What will the project result in?*
* *Who are interested in the outcome of the project?*
* *Who will use the solution?*
* *How will the outcome of the project be used?*
* *...*

*Add a UML use case diagrams that visualizes how the end users will use the website. An example of this is shown in Figure 1 below. To learn more about use case diagrams, see* [*https://creately.com/blog/diagrams/use-case-diagram-tutorial/*](https://creately.com/blog/diagrams/use-case-diagram-tutorial/)*.*



Teacher

Ping Pong



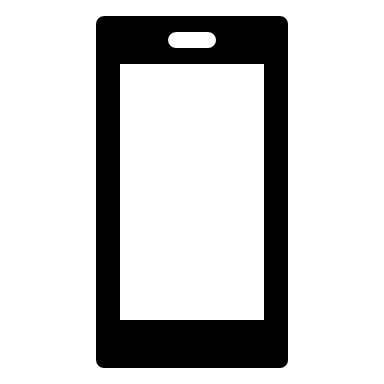
Student

Figure , Use case diagram showing some of the functionality on Ping Pong.

*After having read this chapter, those that have never heard of the project before should a have a good understanding of what it is about. If they would like to learn how it has been implemented, they just need to continue to read the rest of the report.*

# Architecture

*Give an overview of the components your solution consists of (web application, database, web browsers, end-users, etc.) here. Visualize this in a figure and show how the different components make use of each other. An example of this is shown in Figure 2 below. Give also a brief description of each component (the role of the component/why the component is part of the solution).*



Android Chat Application

Database

User A

Use

Runs

SQL

Android Smartphone

Figure , Example of an architecture for a solution where users can chat with each other using  
their Android phones that store the chat messages in a remote database.

*After having read this chapter, the reader should have a broad (but shallow) understanding of how the solution has been implemented.*

# Database

*Give a detailed description of your database, such as what type of database you use and what type of data (your resources) you store there, constraints you have, etc. Also visualize this in an ER diagram. An example of this is shown in Figure 3 below. See* [*https://www.lucidchart.com/pages/er-diagrams*](https://www.lucidchart.com/pages/er-diagrams) *to learn more about ER diagrams.*

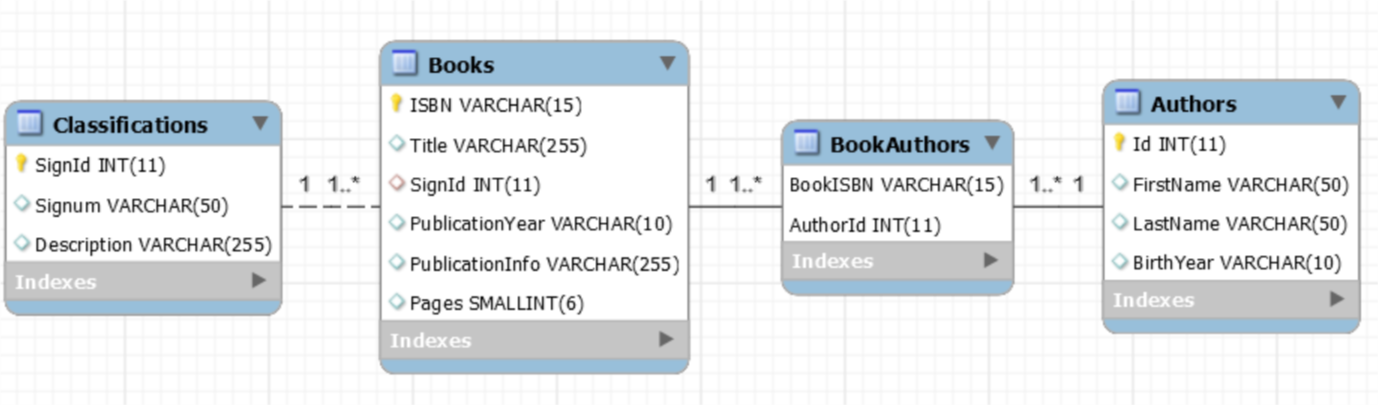


Figure , Example of an ER diagram over a database used to store information about books and authors in a library.

*After having read this chapter, the reader should have a very good understanding of how the data (the resources) on the website is stored. If the reader is a new programmer that should start working on the website, she should now know what she needs to do if she wants to change the resources or add more type of resources (e.g. know how to add a new table to the database with a relation to an existing table).*

# Web Application

*Describe implementation of the web application in detail. Which language have you used? Which framework have you used? Which libraries/packages have you used, and for what purpose? Has all code been written in one file? Or have you somehow structured it in multiple files? How? Are you using some design patterns (e.g. MVC)? Are you using middlewares? Which security vulnerabilities have you thought of and prevented? Etc...*

*You do not necessarily need to show any code to describe the implementation, but feel free to do that if that improves the quality of the report.*

*This chapter will be quite big, so use sub-chapters to structure the content. The template does at the moment contain one subchapter (Graphical User Interface), but you will need more.*

*After having read this chapter, the reader should have a very good understanding of how the web application has been implemented. If the reader is a programmer who should start working on the web application, she should now know where to start when she should implement new features on the web application.*

## Graphical User Interface

*Describe the graphical user interface. Initially this can simply be a low-detailed picture showing all webpages on the website and how they are linked, as shown in Figure 4 below (Note: the click at the top is on the "Contact" link, and not the "Home" link (optical illusion)).*

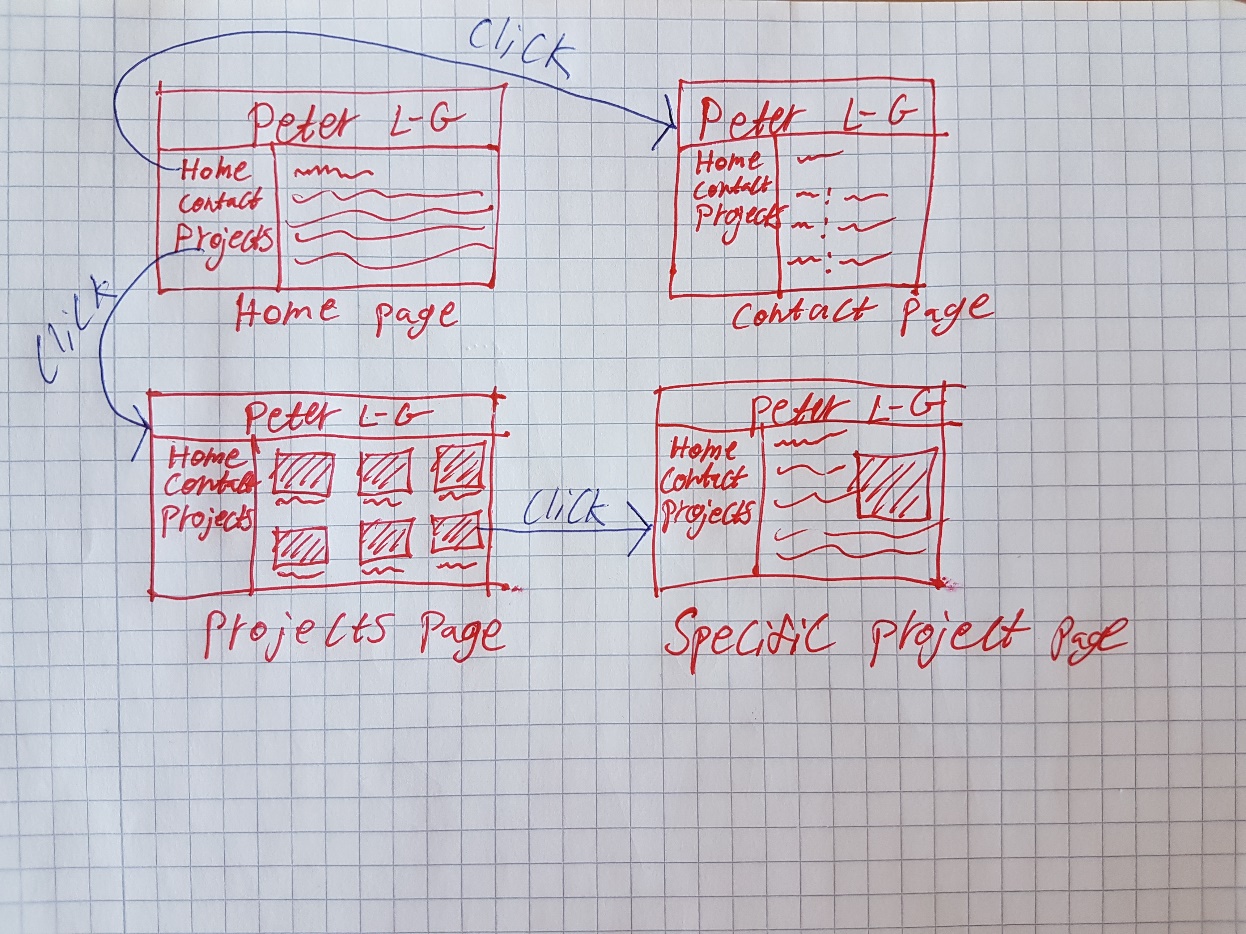


Figure 4, Example of a low-detailed mockup showing the GUI of Peter's personal website.

*When you have implemented the website, showing some screenshots of it here is a very good idea. Keep your wire-frames even when you have real screenshots, they give a good/quick overview of the GUI.*

*After having read this sub-chapter, the reader should understand how the end users will be able to reach their goals through the graphical user interface.*