**Erik Sveningsson** <sver18ed@student.ju.se>  
**Andreas Malmström**<maan1686@student.ju.

A Project Work in *Web Development – Advanced Concepts*

Jönköping University 2020

Eventriloquist

*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 platform 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 (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!*

*This page should of course be removed.*

Table of Contents

[Introduction 3](#_Toc535233012)

[Architecture 4](#_Toc535233013)

[Database 5](#_Toc535233014)

[Web Application 6](#_Toc535233015)

[Single-Page Application 7](#_Toc535233016)

# Introduction

In this project work the main idea is to implement a platform in the shape of a web site. It will be a Single-Page Application that focus on events and recreational activities, similar to a traditional calendar. The drawback with using traditional calendars is that they will become cluttered very fast, and feature a lot of different things, a traditional calendar is also only visible to the person who owns it. The purpose of this web application is to create a sort of interactive calendar online, that only will focus on recreational activities. Users can use the application to plan activities with friends. In a way it will function very much like an online community.

Visitors will be able to register an account on the site and from there create different activities in a calendar, and share them through a message board. Other signed in users will be able to read threads and respond to them, and sign up for activities. Users that has responded to events will then see these events pop up in their calendar on the web site.

This will be useful to people who for example usually try to plan things with their friends, but feel that it sometimes is hard to get everyone on the same page, and want to decide what to do without misunderstandings.

It can also be useful for people who wants to connect with new people, who are also using this web site. They can simply start a thread on the message board and tell people about what they feel like doing, when and where they feel like doing it, and ask other users to join.

# Architecture

*Give an overview of the components the platform consists of (web application, database, web browsers, SPA, end-users, etc.). Visualize this using a figure and show how the different components make use of each other.*

*Are you using Docker? Then it also makes sense to describe which parts of the platform that run in which containers in this chapter. Maybe visualizing this with a figure is a good idea?* ***Hint: Yes, it is****.*

*After having read this chapter, the reader should have a broad (but shallow) understanding of the platforms internal components and structure.*

# Database

*Describe the resources on the platform in detail. What attributes do they consist of? How are they related? Maybe an ER diagram is suitable to have?* ***Hint: Yes, it is.***

*After having read this chapter, the reader should understand how the data on the platform is stored and structured. 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 a resource or add more resources (e.g. know how to add a new table to the database with a relation to an existing table in the database, etc.).*

# Web Application

*Describe implementation details of the web application. 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? Or layers? Are you using some design patterns (e.g. MVC)? Are you using middlewares? Etc... Maybe visualizing this using figures is a good idea?* ***Hint: Yes, it is.***

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

*Does your web application expose a REST API? Then you should also specify how others can use the resources on the application through the REST API. Mention all details about the REST API others needs to know to use it. Maybe describing that in another main chapter is a good idea?*

*After having read this chapter, the reader should have a very good understanding of how the web application has been implemented and how the code is structured. 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 to the web application.*

# Single-Page Application

*Describe implementation details of the Single-Page Application.*