OULU UNIVERSITY OF
APPLIED SCIENCES

Phuong Nguyen

BUILDING AN EVENT PLANNING WEB APPLICATION USING YII2 FRAMEWORK

BUILDING AN EVENT PLANNING WEB APPLICATION USING YII2 FRAMEWORK

Phuong Nguyen
Thesis
Autumn 2018
Business Information Technology
Oulu University of Applied Sciences

ABSTRACT

Oulu University of Applied Sciences
Business Information Technology

Author(s): Phuong Nguyen
Title of Bachelor´s thesis: Building an event planning web application using Yii2 framework
Supervisor(s): Pekka Ojala
Term and year of completion: Autumn 2018 Number of pages: 34

The purpose of this thesis was to build an event planning web application using Yii2 framework to demonstrate the author's learning process and the compatibility of the framework in web development. The application is where users can personally create and manage contents of events and tasks according to their needs.

The application was built based on a PHP framework which provided a useful code generation system with high functional views, models, controller making the development effective and scalable. The thesis will provide a picture how Yii2 framework featuring PHP can make web application development less work effort by using Yii2's featuring tools in a consistent manner. Generally, it has proved that the framework is a good choice for a quick web application development and makes sure it performs well.

The project source code discussion is taken apart in this project thesis and delivered via Github at the following address https://github.com/nguyenhuyphuong2603/eventplanner_thesis

---

Keywords: Yii2, PHP framework, web application development

TABLE OF CONTENTS

VOCABULARY

Yii2	An open source, PHP framework
MVC	Model View Controller
PHP	Scripting language, PHP: Hypertext Preprocessor
OOP	Object-oriented programming
MySQL	Open source relational database management system
NPM	Node Package Manager
CRUD	Create Read Update Delete

1 INTRODUCTION

The purpose of this thesis is to demonstrate the author's learning process and capability of applying a web development framework (specifically, Yii2 framework) to build a web application. The thesis is author's initial idea to build a web application in which users can create events and manage them as they want. The product is developed based on PHP for the core technology, MySQL for the database with the assistance from MySQL Workbench for a better visual database architect, and HTML, JavaScript, CSS as both backend and frontend tool.

Throughout the thesis work, the essential features of the framework have been introduced such as Gii – the code generator which makes the developers less stressed about the code writing, widgets which give a different way to approach to the views. In term of methodology, a large amount of information has been gained mostly via the framework's official documentation and guide. Educationally speaking, the work has given and widened the author's opportunities to have a clear and practical vision to apply theoretical educational background into a real-life application.

The thesis explains the process of application design and its implementation utilizing the framework's outstanding features to illustrate its advanced abilities and effectiveness. Specifically, taking advantages of Yii brings a great deal of outcomes to build a CRUD operational application which has been proved throughout the development process.

The thesis work's result is a properly functional web application that has been reached in terms of capability of running on a localhost and after a server. The application's expectation has been met for a web application where end users can perform actions they require to use as an online planning event tool.

2 THE YII2 FRAMEWORK

2.1 Introduction to Yii2 Framework

2.1.1 OOP PHP

Object-oriented programming (OOP) is a programming language added to PHP 5 which makes OOP an approach to software development using objects, which include attributes, and methods in building web application reusable and easier. Object-oriented programming wraps all the concepts e.g. class, object, polymorphism, encapsulation, abstraction, inheritance.

PHP is an object-oriented scripting language and supports all the OOP major principles, it is a strong OOP language which was chosen for this project development. Moreover, a PHP framework was considered to inherit and benefit from all the outstanding PHP features to build a reliable, fast web application.

2.1.2 Yii2 Framework

According to Portwood II (2016, preface), Yii2 framework is an open source, full-stack and high-performance component-based PHP framework primarily designed for building modern, scalable web applications using PHP as well as RESTful API web services, and so on. Furthermore, it has several code generation facilities and create-read-update-delete (CRUD) interface maker. The framework is fully supported in documentation, guide, and APIs as well as a large actively contributed community.

Yii2 framework implements the model-view-controller (MVC) architectural design pattern like most of other popular PHP frameworks. The framework seizes latest technologies for web development such as composer, namespaces. As a result of fact, namespaces introduction in Yii2 gives opportunities that developers

encounter solving big problems of name collisions between written code and internal PHP classes/functions or third-party libraries. In additional, using composer is preferred because it enables developers to install, update, and manage all dependencies and extensions for applications by running a single command. It also keeps applications kept up to date with the latest security and bug fixes. (The Definitive Guide to Yii 2.0b, cited 15.02.2018.)

Currently, Yii2 framework introduces two available templates to download, advanced template and basic template which was implemented in this thesis application development. Besides implementing MVC, Yii2 framework introduces a front-controller, called Application, encapsulating the execution context for the processing of a request.

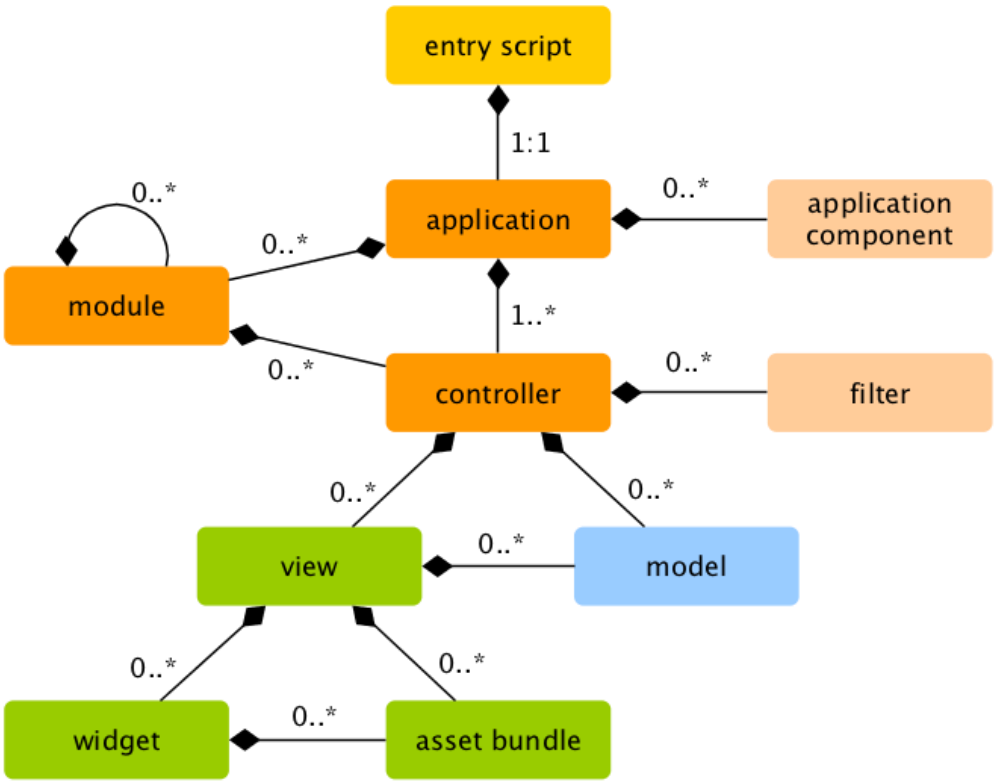

*FIGURE 1. A typical static structure of an application (*The Definitive Guide to Yii 2.0a*, cited 15.02.2018.)*

Moreover, there are different following entities in Yii applications listed below

- Entry scripts: Accessible PHP scripts responsible for starting a request handling cycle
- Applications: Globally accessible objects that manage application components and coordinate them to execute requests
- Application components: registered objects with applications and provide services to fulfil requests
- Modules: self-contained packages contain complete MVC by themselves
- Filters: code that need to be invoked before and after the actual handling of requests by controllers
- Widgets: objects embedded in views

(The Definitive Guide to Yii 2.0a, cited 15.02.2018.)

2.2 Code Generation With Gii

Gii is provided in Yii as a module. Gii can automatically generate code that implements some common Web site features. Using Gii to automatically generate code is a matter of right input information per instructions. (The Definitive Guide to Yii 2.0d, cited 24.05.2018.)
Gii is preferably installed via composer with this command (Yii2 Framework 2018, Tool Gii. Cited 24.05.2018.)

```
composer require “yiisoft/yii2-gii:*”
```

Once it has been installed, Gii can be configured in the modules property of the application provided in the config/web.php file to set the module named gii included when in a development environment.

```
$config = [ … ];
If (YII_ENV_DEV) {
        $config[‘bootstrap’][] = ‘gii’;
        $config[‘modules][‘gii’] = [
```

```
            'class' => 'yii\gii\Module',
        ];
    }
```

The above configuration states that Gii has been enabled and ready to be accessed via the URL: http://hostname/index.php?r=gii (The Definitive Guide to Yii 2.0d, cited 24.05.2018.)

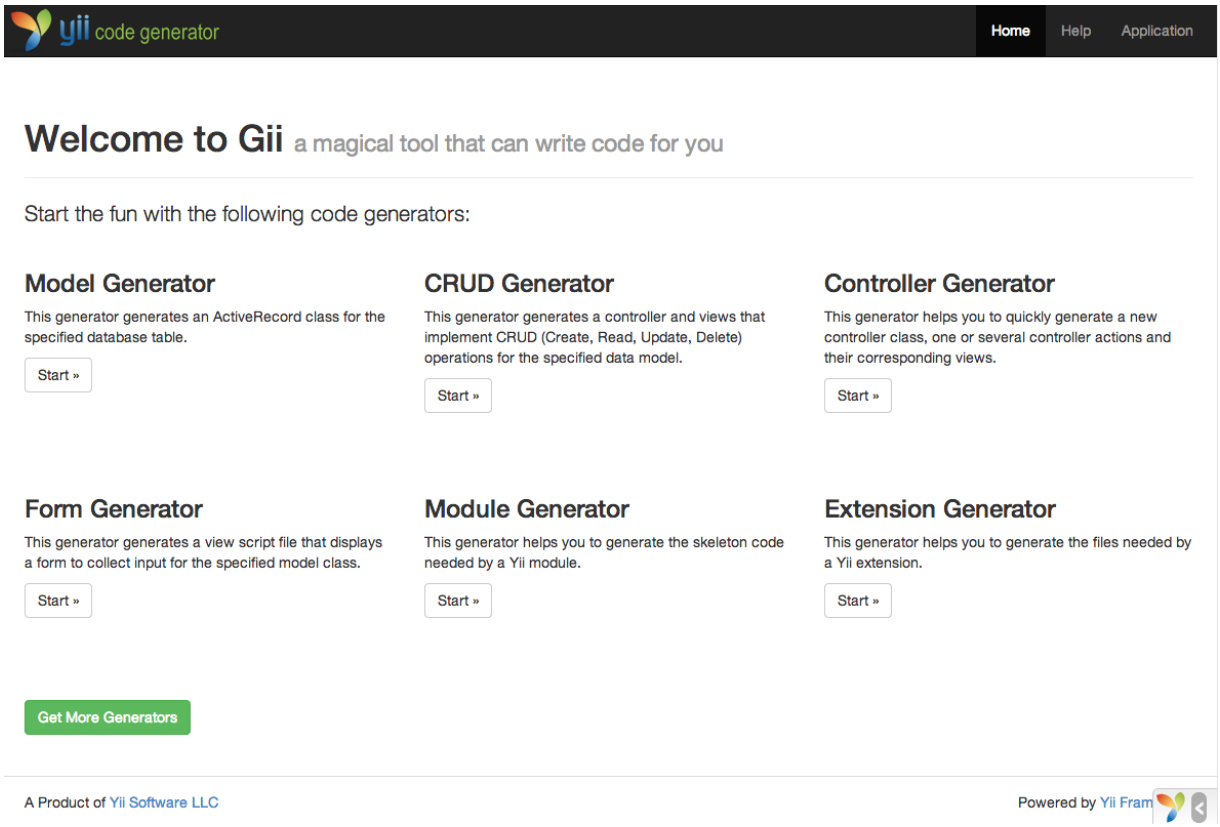

*FIGURE 2. Gii code generator entry page (*The Definitive Guide to Yii 2.0d*, cited 24.05.2018.)*

Yii2 is famous for its convenient tool, which is Gii, that gives developers a quick access to commonly used code snippets and complete CRUD controllers. There are some available default generators as below.

- Model Generator: generates an ActiveRecord class for a detailed database table
- CRUD Generator: generates a controller and views that implement CRUD operations for the specified model

- Controller Generator: generates a controller class, one or numerous controller actions and related views
- Form Generator: generates a view script to display a form to collect input for the specified model class
- Module Generator: generates skeleton code needed by a Yii module

(Yii2 Framework 2018, Tool Gii 2018, cited 24.05.2018.)

2.3 Active Record

Active record accommodates an object-oriented interface for connecting and manipulating data stored in databases. An active record class is mapped with a specified database table, an active record instance corresponds to a row of that related table, and an attribute of an active record instance represents the value of a specified column in that row. Active record instances are preferred as models. That is the reason why active records classes are placed under app\models namespace. For this reason, yii\db\ActiveRecord inherits from yii\base\Model all model features e.g. attributes, validation rules, etc. (The Definitive Guide to Yii 2.0e, cited 18.06.2018.)

Database connection can be configured in the application configuration like in the following example:

```
return [
    'components' => [
        'db' => [
            'class' => 'yii\db\connection',
            'dsn' => 'mysql:host=localhost;dbname=testdb',
            'username' => 'example',
            'password' => 'example',
        ],
    ],
];
```

Rather of raw SQL statements, active record attributes can be accessed, and active record methods are able to be called to access and manipulate the data in database tables.

The following relational databases are supported by Yii active record:

- MySQL 4.1 or later via yii\db\ActiveRecord
- PostgreSQL 7.3 or later: yii\db\ActiveRecord
- SQLite 2 and 3: via yii\db\ActiveRecord
- Microsoft SQL Server 2008 or later: via yii\db\ActiveRecord
- Oracle: via yii\db\ActiveRecord
- CUBRID 9.3 or later: via yii\db\ActiveRecord
- Sphinx: via yii\sphinx\ActiveRecord
- ElasticSearch: via yii\elasticsearch\ActiveRecord

Moreover, Yii also supports using Active Record with NoSQL databases:

- Redis 2.6.12 or later: via yii\redis\ActiveRecord
- MongoDB 1.3.0 or later: via yii\mongodb\ActiveRecord

(The Definitive Guide to Yii 2.0e, cited 18.06.2018.)

2.4 Widgets

Widgets are reusable client-side code, which contains HTML, CSS, and JavaScript primarily used in views to create complex and configurable user interface elements in an object-oriented manner. A widget can be called by function yii\base\Widget::widget() to apply in a view as it can be seen like shown under (The Definitive Guide to Yii 2.0f, cited 27.06.2018.)

```
<?php
Use yii\jui\DatePicker;
?>
<?=DatePicker::widget([
        ‘model’ => $model,
        ‘attribute’ => ‘from_date’,
        ‘language’ => ‘ru,
        ‘dateFormat’ => ‘php:Y-m-d’,
]) ?>
```

There are several widgets require a block of content which it is needed to be enclosed between yii\base\Widget::begin() and yii\base\Widget::end() functions (The Definitive Guide to Yii 2.0f, cited 27.06.2018.). This can be inserted in a view like the following example:

```
<?php
use yii\bootstrap\ActiveForm;
use yii\helpers\Html;
?>
<?php $form = ActiveForm::begin([‘id’ => ‘login-form’,
‘enableClientValidation’ => false]);
?>
<?= $form
        ->field($model, 'username', $fieldOptions1)
        ->label(false)
        ->textInput(['placeholder'=>$model->getAttributeLabel('username')])
   ?>
<?=  $form
          ->field($model, 'password', $fieldOptions2)
          ->label(false)
          ->passwordInput(['placeholder'=>$model-
          >getAttributeLabel('password')])
   ?>
<div class=”col-xs-4”>
              <?= Html::submitButton('Sign in', ['class' => 'btn btn-primary btn-
block btn-flat', 'name' => 'login-button']) ?>
 </div>

<?php ActiveForm::end(); ?>
```

The above usage of Widgets is directly taken out of from the product source code to demonstrate the flexibility of Widgets

3 DESIGN AND IMPLEMENTATION

3.1 Development Environment

3.1.1 Installing Yii

There are two most popular ways to install Yii2, downloading the framework from source control (typically, from GitHub at https://github.com/yiisoft/yii2) or using the composer package manager-a package dependency management tool for PHP (Portwood II 2016, 1.).

The first option is downloading using composer package. When the composer is installed it can be used by using command line. During the application development, all the software installed are stored in the vendor folder to keep track with. Composer can be installed by following the instructions on https://getcomposer.org. Portwood II (2016,3.) stated that once composer is installed, a global plugin called the composer asset plugin is required to install (available at https://github.com/francoispluchino/composer-asset-plugin). This plugin enables composer to manage asset files without the need to install additional software. Composer global require "fxp/composer-asset-plugin:1.0.0" with composer installed, after that Yii application basic template is ready to download by the following command.

```
composer create-project --prefer-dist yiisoft/yii2-app-basic basic
```

The above command will have the Yii2 basic app installed to a folder named basic. The create-project command is recommended when a new project is started to clone “yii2-app-basic” for development. However, a Yii2 project can be also created from scratch and it will be more complicated but gives developers more control over the application’s structure (Portwood II 2016, 4.).

The other way to install Yii2 is done by installing from source control. This process includes three steps as listed following:

- Download the archive file from the official website at yiiframework.com
- Unpack the file to a Web-accessible folder
- Modify the config/web.php file with a secret key for the cookieValidationKey configuration item. (The Definitive Guide to Yii 2.0c, cited 25.03.2018.)

Importantly, PHP installation should be configured in order to meet the minimum requirements of Yii. PHP 5.4 or above is required, ideally latest PHP 7. There is a built-in requirement script called requirements.php used for checking values to make sure Yii2 is able to run on server (local or remote) (Portwood II 2016, p. 7).

```
php requirements.php
```

Error message can be seen in check conclusion section after the command script, if there is no error alert, the application is ready to move forward. Yii 2 basic project template directory structure looks like the following after everything is done with the installation.

```
assets/        contains assets definition
  commands/        contains console commands (controllers)
  config/       contains application configurations
  controllers/     contains Web controller classes
  mail/        contains view files for e-mails
  models/        contains model classes
  runtime/       contains files generated during runtime
  tests/       contains various tests for the basic application
  vendor/        contains dependent 3rd-party packages
  views/       contains view files for the Web application
  web/       contains the entry script and Web resources
```

FIGURE 3. A basic project template directory structure (Yii 2.0 Basic Application Template 2018, cited 25.03.2018.)

3.1.2 Working with databases

Yii will not create the database by itself, this process must be done manually before accessing it. Therefore, in this project MySQL Workbench 6.3 CE was used in order to create database for the application and also tables' relationships, diagrams as well. Keeping Yii2 function properly required MySQL 4.1 or later.

During the application development, there were four database tables created with necessary information. They were user table, profile table, event table and task table. User table contained authenticated information of user when signing into the application such as username and password.

TABLE 1. User database table

Attribute	**Description**	**Type**
user_id	User id, primary key	INT(11)
username	Username for user, it can include characters and numbers	VARCHAR(100)
password	User password to access	VARCHAR(100)

The second table was profile table, including user registered information when a new account created. All the information e.g. first name, last name, e-mail, was given by new user and stored in this table.

TABLE 2. Profile database table

Attribute	**Description**	**Type**
id	Primary key, profile id	INT(11)
firstname	First name of user when registered	VARCHAR(255)

lastname	Last name of user when registered	VARCHAR(255)
email	User email	VARCHAR(255)
birthday	Date of birth	DATE
city	City of user's choice	VARCHAR(255)

The other two tables were event and task tables where content of event and task was created by users. Both tables consisted of some common attributes due to parental relationship between them. In this case, event table was parent table of task table. Event database table could be seen in following table

TABLE 3. Event database table

Attribute	**Description**	**Type**
event_id	Primary key, event id	INT(11)
event_date	Date of event, chosen by user	DATE
creator_id	Creator id, user id	INT(11)
reminder	Boolean. Set reminder for the event. Value 0 means no reminder	TINYINT(11)
description	Description for the event	TINYTEXT
name	Event name	VARCHAR(150)

TABLE 4. Task database table

Attribute	**Description**	**Type**
task_id	Primary key, task id	INT(11)
description	Description text area for the task	TINYTEXT
event_id	Foreign key to Event table, to keep track the relationship	INT(11)

duedate	Due date for the task, set by user	DATE
reminder	Boolean, Set reminder for the task. Value 0 means no reminder	INT(1)
Parent_task_id	Keep track relationship with parent table	INT(11)
Creator_id	User id	INT(11)
note	Text note	TINYTEXT
name	Task name	VARCHAR(50)

Those tables were stored in database and could be simply explained by the diagram below

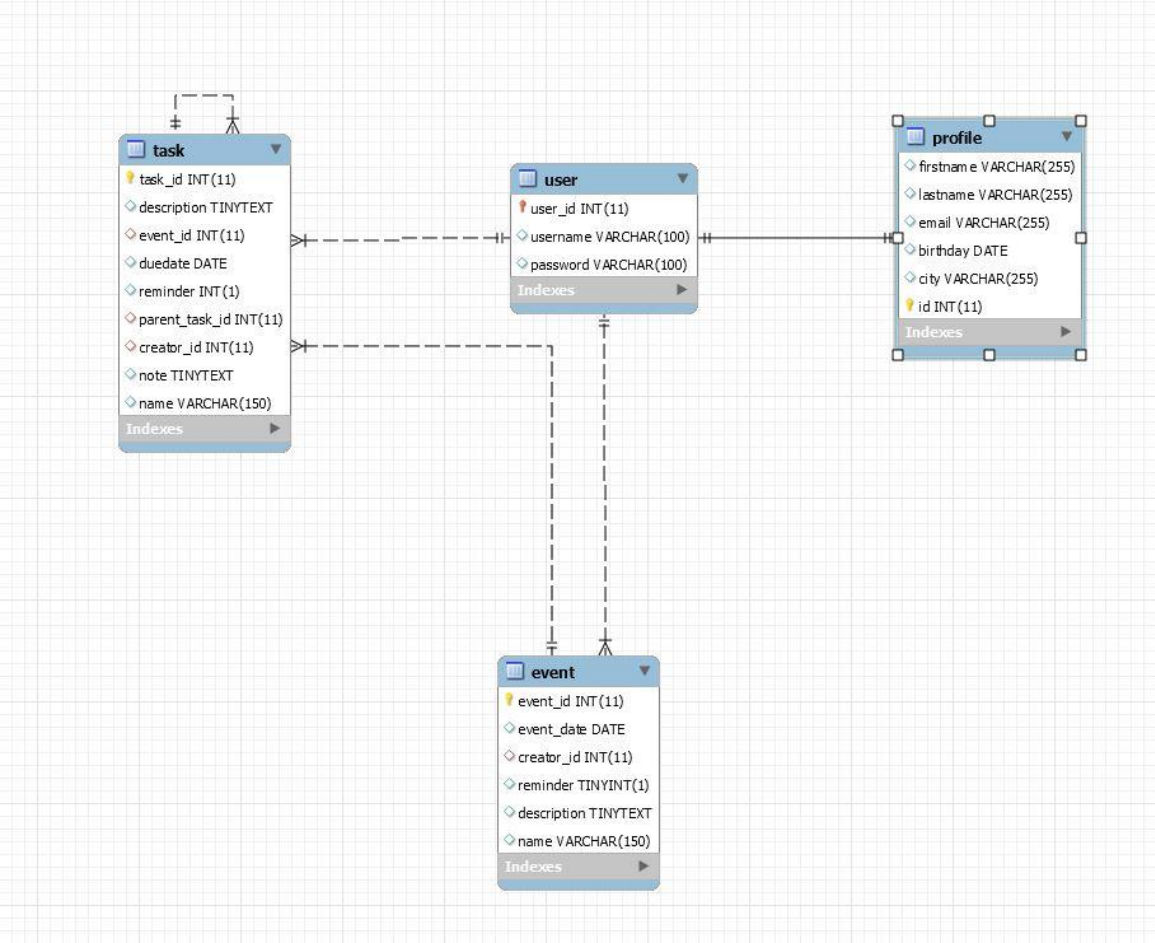

FIGURE 4. Database table diagram

User table had one-to-one relationship with profile table and one-to-many relationship with event table and task table. While event table had zero-to-many relationship with its child task table.

3.2 Design

The event planner web application project firstly was a personal idea in making a web application for personal use and technically expandable for other users' expectations. Initially, the web application was designed to be fulfilling some basic requirements for end users such as create content, set day time, modify, and delete event or task. Theoretically, reminder feature was also considered to be implemented to send reminder e-mail to users when the due date is coming.

The user interface was primarily designed to be user-friendly and easy to use. All the basic functions and requirements were introduced on the application e.g. logging, main dashboard where users allow to manage content and events, users also can create new tasks, edit and delete unwanted ones. The web application was done at first on paper sketch and finally done by using web application Pencil Project at https://pencil.evolus.vn/. These are following sketches were made from the designing stage:

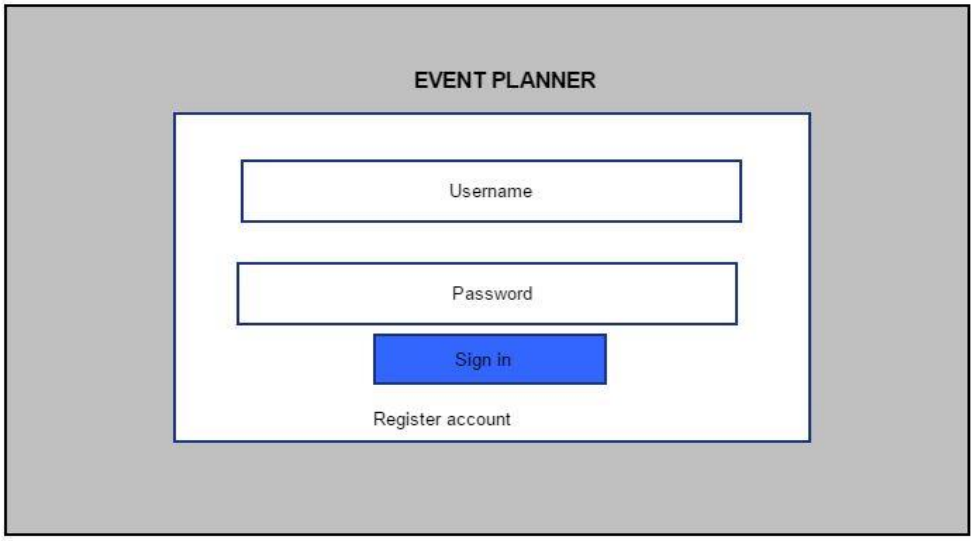

FIGURE 5. Prototype log in page for the application

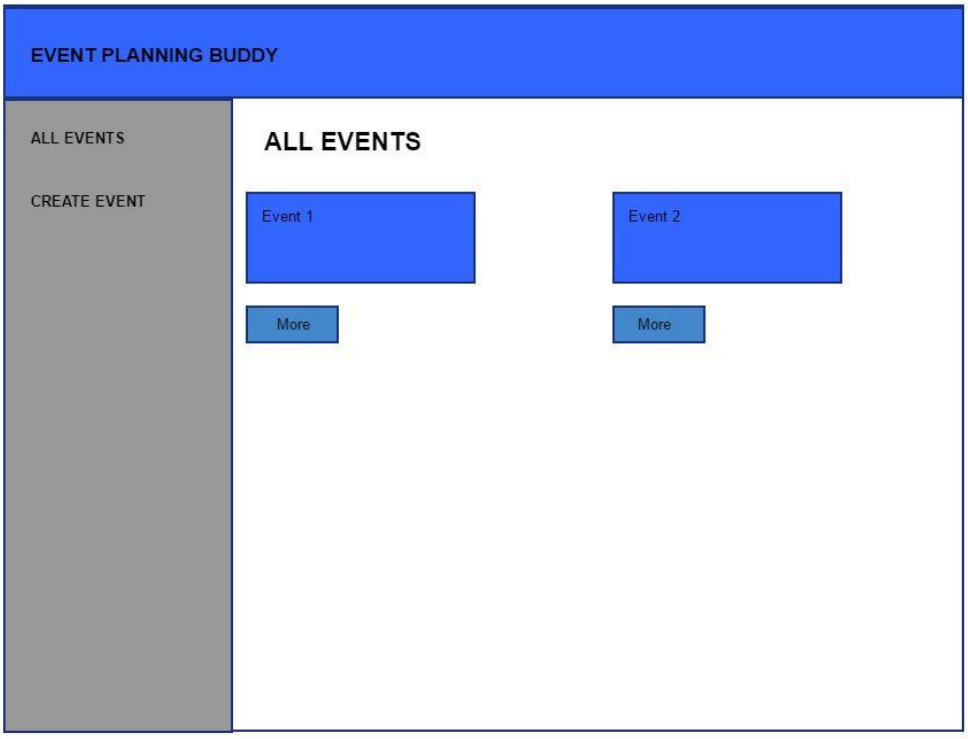

FIGURE 6. The main dashboard event manager

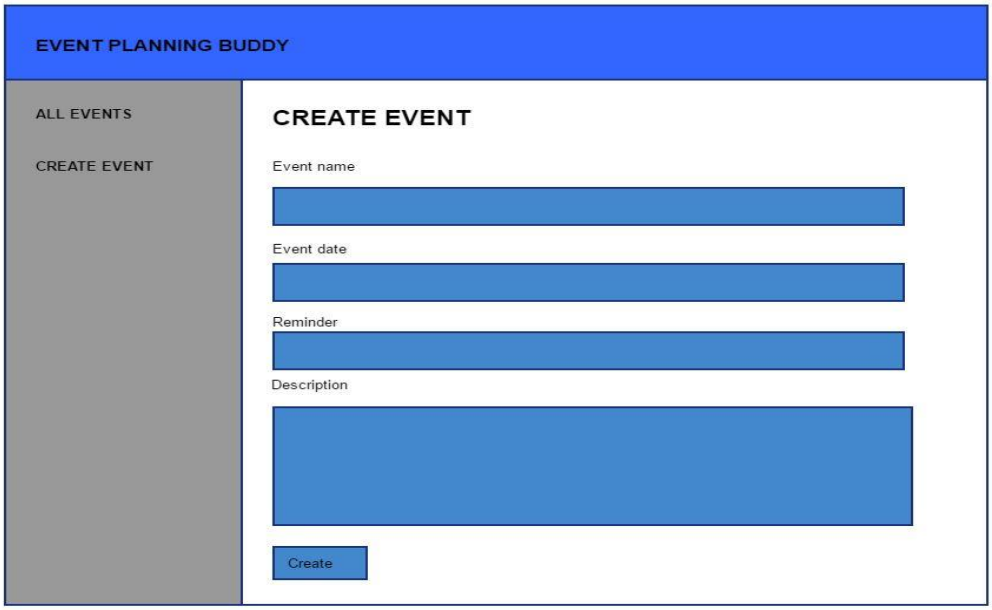

FIGURE 7. Event creation page

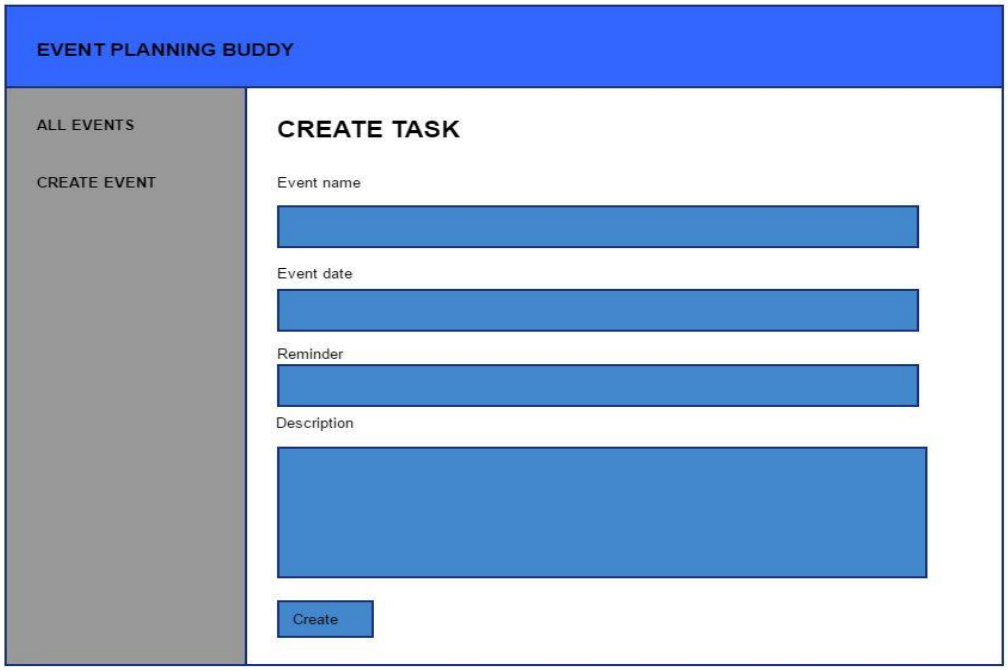

FIGURE 8. Task creation page

3.3 Implementation

This event planning web application requires user validation, membership will be asked for before using the application. Users are able to save the content and keep track of what they have created in order to accomplish the event and/or the task. Finally, logging out of the application when end users are finished.

3.3.1 Register

A new account is required to start using the application, register form will be displayed to membership sign up. Site controller involves primarily in this step with action “actionRegister()” to validate if user is a guest. Model “RegisterForm” will be initialized when a guest creates a new membership.

TABLE 5. Information of new account register action

Action	**Description**	**Return**
actionRegister()	In this action, the register form will be shown when user is not a guest. Otherwise, user will be redirected to homepage. To register, user need to fill the register form and submit it to server. This action will check the Http Post request from the form and insert new user and its profile into database based on information from request.	Render view of register form for membership creation when model is not submitted Redirect to welcome page where new event can be created

3.3.2 Authentication

Log in

Site Controller is responsible for application log in process and action “actionLogin()” is the main method for this. The application home page is routing to ‘site/login’ under actionIndex(). If user is an existing member, the authentication login form will be delivered to verify username and password. The process will return to the home page if the verification is correct, unless the login page will stay remained

TABLE 6. Log in action in Site controller

Action	Description	Return
actionLogin()	In this action, the login form will be shown when user is not guest. Otherwise, user will be redirected to homepage. To login, user need to fill the form and submit it to server. This action will check the Http Post request from the form and validate, retrieved user and its profile based on username and password come with the request.	After logging in, redirect to home page. The homepage will show list of events which have been created by logged user. In case there are no events created by logged user. There will be button to create event for users to create easier.

Log out

Log out action is included in site controller which is the default controller for the application. Action “actionLogout()” is called when users log out of the application and the view is redirected to the log in page.

TABLE 7. Log out action in Site controller

Action	Description	Return
actionLogout()	Log user out of the application Logs out the current user. This will remove authentication-related session data.	Redirect to the login page

3.3.3 Features

This part mentions about the important features which the application is built upon. The purpose of this application is to allow user to create, edit, and even delete events and/or tasks during the experience. The event and task will be discussed in the following section, while event is going to be be used without a single task, tasks on the other hand will not function without being belonged to an event.

Event

Events can be created, modified and deleted from user’s event main page. Event controller is involved using actions like “actionCreate()” to create new event,

“actionUpdate()” to edit an existing event, and “actionDelete()” to delete an event from application database permanently.

After an event is created, an automatic e-mail will be sent through action “sendMailToCreator()” in event model.
More detailed information can be seen in table.

TABLE 8. Event controller

Name	**Description**	**Return**
actionCreate()	Create a new single event follow the information which has been included in the Http Post request. An e-mail was sent to user’s register mail to confirm a new account had been made	Render view with form for inputting data when model is not submitted When submitted- data submitted and redirected to view page.
actionUpdate()	Modify an existing event. An event will be retrieved from database based on the event_id which has been passed in the route /event/update/{$event_id}	If submitted, database added to database table and view will be redirected to view page
actionDelete()	Delete an event out of the application database An event will be retrieved from database based on the id which has been passed in the route	If deletion is successful, the browser would be redirected to the 'index' page.

	/event/delete/{$id}. System will throw NotFoundHttpException and user will be redirected to 404 page when cannot find event. Otherwise, event will be removed and user will be redirect to page which show all events.	

Task

Familiar to event section, tasks can be created, edited, and deleted out of the application database as well. All the methods and actions are taken place in task controller using actions such as “actionCreate()” to create a new task, “actionUpdate()” to edit and “actionDelete()” to get rid a task out of the database system.
More detailed information about task can be viewed through the below table.

TABLE 9. Task controller

Name	**Description**	**Return**
actionCreate()	Create a new single task. This function will insert new task into database and assign it to an event based on the event_id which is included in the http GET request.	Render view with form for inputting data when model is not submitted When submitted- data submitted and browser would be redirected to view page.
actionUpdate()	Modify an existing task.	If submitted, database added to database table and browser is

	This function will retrieve task based on the id route parameter which has been past to server through route /task/update/{$id}.	redirected to view page
actionDelete()	Delete an event out of the application database A task will be retrieved from database based on the id which has been passed in the route /task/delete/{$id}. System will throw NotFoundHttpException and user will be redirect to 404 page when it cannot find the task. Otherwise, task will be removed, and user will be redirect to the view page of its event.	If deletion is successful, the browser would be redirected to the view page.

3.3.4 Third-party software

Kartik-v DetailView

An extended version of Yii2 DetailView with additional features functioning effectively in VIEW and EDIT modes supported and provided by Yii2 library. The DetailView extension enabled styling methods for view widgets, forms, columns and improved general usability and various enhancements to the web application. (Kartik-v DetailView 2018, cited 02.07.2018.)

AdminLTE Asset Bundle for Backend Theme in Yii2 Framework

The package bundle contained CSS and JavaScript files for rendering web page view to perform a better usability for the application (AdminLTE Asset Bundle for Backend Theme in Yii2 Framework 2018, cited 05.07.2018.). Boostrap-datepicker was also included in this package and implemented onto the application development.

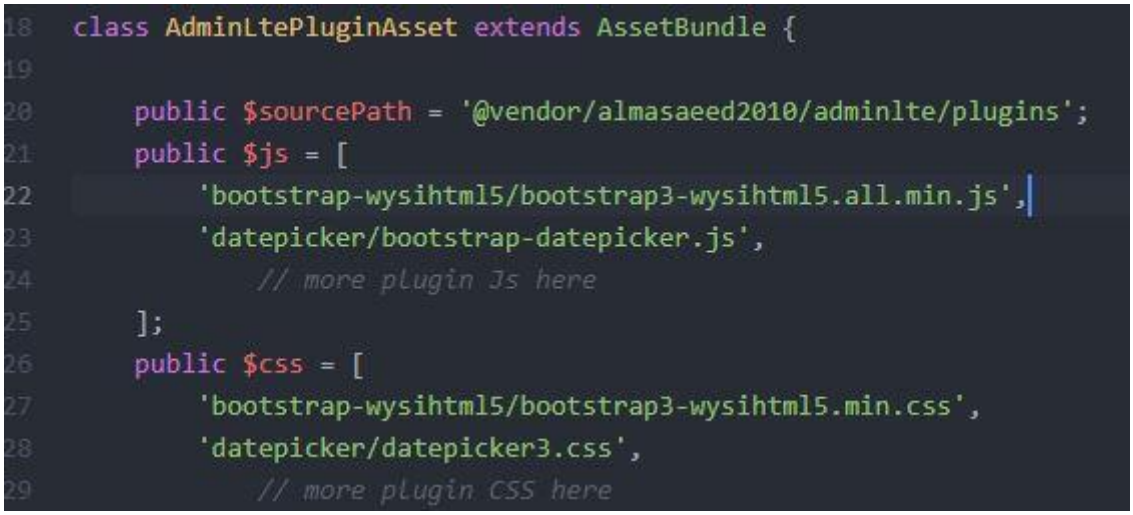

FIGURE 9. AdminLTE Asset Bundle file

4 FINAL PRODUCT

Implementing and using third-party software changed the original user interface which brought a better experience. There were remaining various of styling bugs and imperfect choice of design, but the application run smoothly with required features mentioned in the beginning of the product development. The further development would need to be invested to extend and add more other better features and functions.

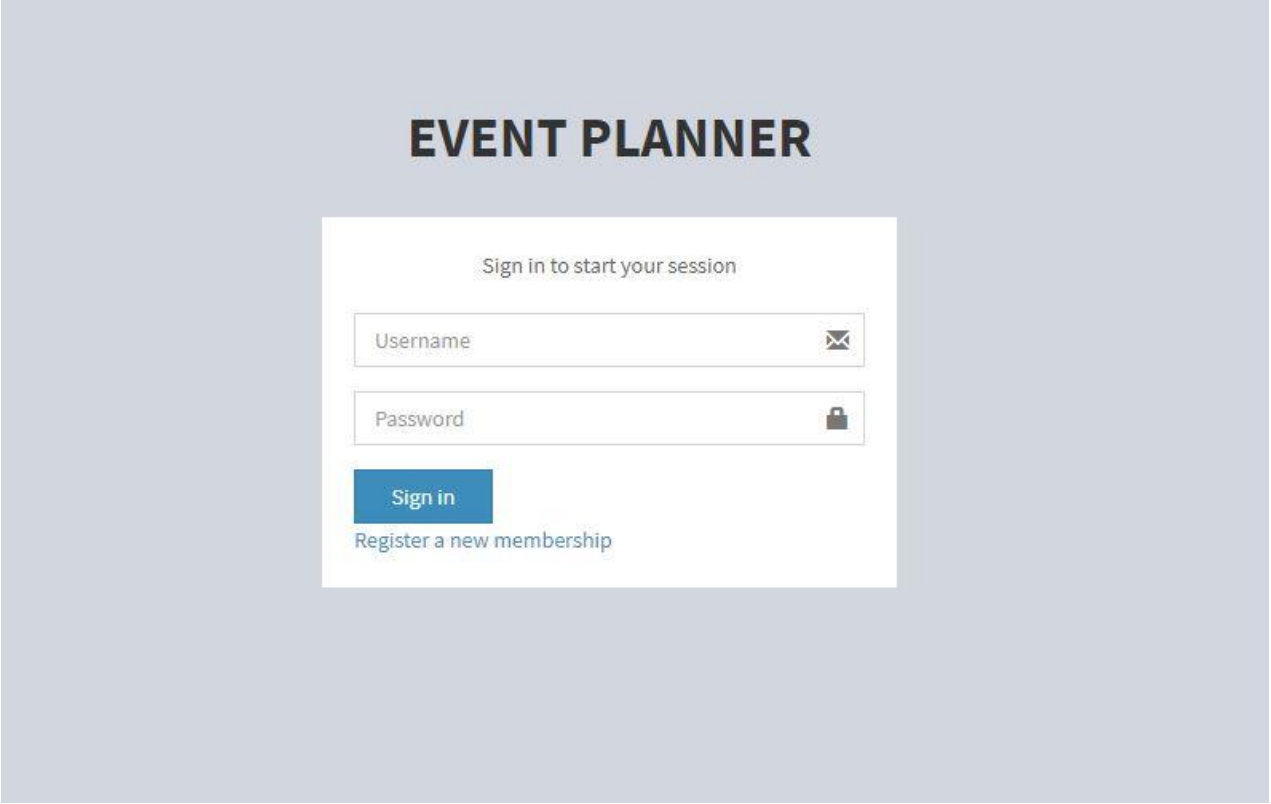

FIGURE 10. The main log in page

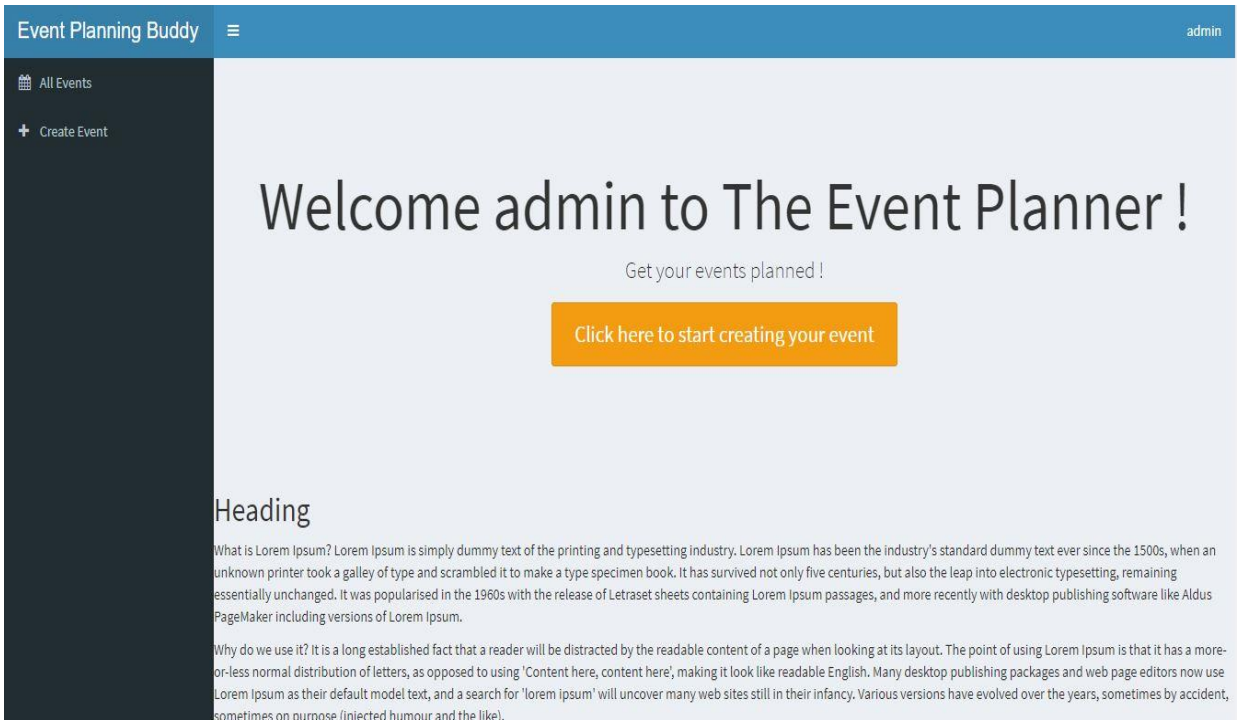

FIGURE 11. Welcome page for new user

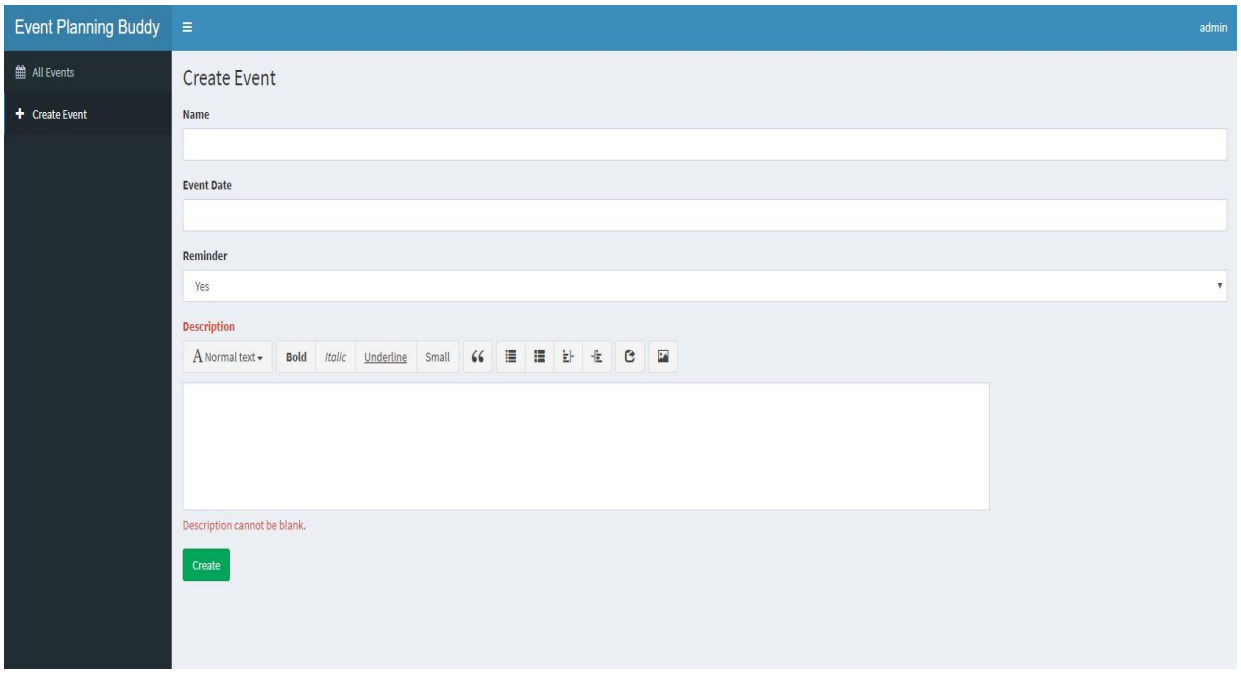

FIGURE 12. Create event view

5 CONCLUSION

The event planning web application now can run and function all the features of creating events and tasks as required. Yii2 framework has proved its ability and strength in building CRUD web applications through providing necessary components for the application development. During the long process of building the application, Yii has been performing properly and encountering issue free about versions or updates. The application compatibility can be maintained by using the package control. The project can be usefully extended with Yii advanced template where developer can separately focus on specific side of choose, backend or frontend. Moreover, a mobile version with same features will be able but some other technologies may be considered because PHP and Yii2 framework are not popular for mobile application development.

Finally, the project has been met its core requirements and use of Yii2 framework has been fully applied to the development. However, the Reminder feature of reminding users on the due date was not implemented due to the complexity. The application will be more functional and impressive with this feature, the feature can automatically send e-mail to remind users about upcoming event's due date. The complexity of the objective is beyond the support of the Yii2 framework and requires a different approach and technology, which makes the feature left out. In the end, from the technical point of view the thesis provides and introduces some abilities and information of Yii2 framework in web development and it can be a choice among other popular PHP frameworks.

6 DISCUSISION

It required a great deal of effort and investment to apply what I have studied at school and self-taught process into a practical project which was my thesis topic. The learning process took a long time and somehow incomplete at some point due to applying practical technologies to a practical product. Before starting the project, my web development and knowledge about related web technologies were limited and vague. It was a challenge to bring a web development framework, Yii2 framework, to my application development. I had to put more time and effort on core principles of PHP before getting to the framework. The process was long and desperate enough to make me want to change my mind sometimes. Using a web development framework was not the difficult thing I found but building an application as you wanted was harder.

My web application finally worked and run smoothly which made me happy and relieved, but I found that it could be done even better and quicker in the future development with up-to-date technologies and other frameworks. The application could be used mostly for basic personal purposes on planning and noting small events. It could be looking better with more elegant styling and user interface design as well as it could have a mobile version with simple functions and features for basic needs from users. However, the change of technologies and frameworks would be considered for the mobile application and that would be considered later.

REFERENCES

AdminLTE Asset Bundle for Backend Theme in Yii2 Framework 2018, AdminLTE Asset Bundle. Cited 05.07.2018, https://github.com/dmstr/yii2-adminlte-asset

Charles R. Portwood II. 2016. Mastering Yii: Advance your modern web application development skills with Yii Framework 2. Cited 03.06.2017, http://proquest.safaribooksonline.com.ezp.oamk.fi:2048/book/web-development/web-services/9781785882425

Kartik-v DetailView 2018, yii2-detail-view. Cited 02.07.2018, https://github.com/kartik-v/yii2-detail-view

The Definitive Guide to Yii 2.0a, Application Structure: Application Structure Overview 2018. Cited 15.02.2018, https://www.yiiframework.com/doc/guide/2.0/en/structure-overview

The Definitive Guide to Yii 2.0b, Introduction: About Yii 2018. Cited 15.02.2018, https://www.yiiframework.com/doc/guide/2.0/en/intro-yii

The Definitive Guide to Yii 2.0c, Getting Started: Installing Yii 2018. Cited 25.03.2018, https://www.yiiframework.com/doc/guide/2.0/en/start-installation

The Definitive Guide to Yii 2.0d, Getting Started: Generating code with Gii 2018. Cited 24.05.2018, https://www.yiiframework.com/doc/guide/2.0/en/start-gii

The Definitive Guide to Yii 2.0e, Working with databases: Active Record 2018. Cited 18.06.2018, https://www.yiiframework.com/doc/guide/2.0/en/db-active-record

The Definitive Guide to Yii 2.0f, Application Structure: Widgets 2018. Cited 27.06.2018, http://www.yiiframework.com/doc-2.0/guide-structure-widgets.html

Yii 2.0 Basic Application Template 2018. Cited 25.03.2018, https://github.com/yiisoft/yii2-app-basic/

Yii2 Framework 2018, Tool Gii. Cited 24.05.2018, https://yii2-framework.readthedocs.io/en/stable/guide/tool-gii/