TALLINN UNIVERSITY OF TECHNOLOGY

School of Information Technologies

Department of Software Science

Kevin Sults 185327IAIB

Development of Disc Golf Event Planner

Bachelor Thesis

Supervisor

Gert Kanter

Doctor of Philosophy
(Computer Science)

Tallinn 2022

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Kevin Sults ....

(signature)

Date: May 29, 2022

Annotatsioon

Lõputöö eesmärk oli kettagolfi võistluste planeerija loomine. Planeerija mahutab endas kasutaja plaanitud võistlusi, kuvab võistlusi toimumisjärjekorras, pakub võimalust saada kiirülevaadet võistlustest ning laiendada neid täisvaateks kogu olemasoleva informatsiooniga.

Lõputöö raames on vajalik ka kasutajaprofiili loomine, kus rakendus jälgib kasutaja harjumusi ning oskab selle põhjal soovitada otsinguparameetreid võistluste otsingusse ja oskab soovitada graafikus olevatesse aukudesse võistluseid puhuks kui kasutaja ise pole veel leidnud sobivaid võistluseid. Tehtud on ka integratsioon Google Mapsiga ning on võimalik visualiseerida kogu oma graafikut rakendusse integreeritud kaardil.

Rakendusel on tehtud integratsioon professionaalse kettagolfi assotsiatsiooniga (ingl Professional Disc Golf Association (PDGA)), mis käitub kui võistluste info algallikas. Planeerija toimib PDGA võistluste vahendajana, seob neid kasutaja personaalse reisiinfoga ning võimaldab ka kasutajal enda planeerija jaoks luua võistlusi nullist.

Tulemuseks on töötav veebirakendus, mis võimaldab kasutajal luua omaenda võistlusgraafikut, jälgida võistlustega seotud informatsiooni ning olla alati kursis eelootava registreerimiskuupäeva ja võistlusega läbi kalendrisündmuste eksportimise. Kasutaja saab isikustatud soovitusi võistluste kohta, saab jälgida oma eesmärki ning seadistada võistlushooaega. Võistlusplaani on võimalik visualiseerida erinevatele kujudele.

Lõputöö jooksul kirjutatud kood on leitav aadressilt https://gitlab.cs.ttu.ee/kesult/iaib.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 28 leheküljel, 6 peatükki, 11 joonist, 0 tabelit.

Abstract

Purpose of this thesis was the development of a disc golf events planning application as a website. The planner maintains events that the user is willing to compete in during the year, displays them in the order they take place in and offers a quick preview of all events in the planner and is expandable to full overview including all available details.

The thesis also includes the creation and calculation of so called user profile, meaning the application is monitoring user habits of which events are being added to and removed from the schedule, enabling web application to recommend events to slots in schedule in case the user is yet to find the correct event to fit its needs. Possibility to prefill search is built upon this as well. User also has the possibility to visualize the schedule on an integrated Google Maps map.

An integration with the PDGA (Professional Disc Golf Association) database via their API has been made, which acts as the source of truth for event information. Planner acts as an intermediary between the API and user, enabling the union of personal travel information with the PDGA event. The user is also able to create fully customized events, hosted in the web application database.

For the result there is the working web application, enabling the user to create a competition schedule for disc golf, bind travel information to events and always be informed of the upcoming events and registration dates with personalized recommendations of potentially interesting events. The plan can be visualized in a few different ways.

Thesis code can be found in https://gitlab.cs.ttu.ee/kesult/iaib.

The thesis is in English and contains 28 pages of text, 6 chapters, 11 figures, 0 tables.

List of abbreviations and terms

AD	Active Directory
AM	Amateur
API	Application Programming Interface
AWS	Amazon Web Services
B2C	Business to Consumer
CGNAT	Carrier-Grade Network Address Translation
CSS	Cascading Style Sheets
DDD	Domain Driven Design
EF	Entity Framework
IDE	Integrated Development Environment
IIS	Internet Information Services
IP	Internet Protocol
JS	JavaScript
JSON	JavaScript Object Notation
JWT	JSON Web Token
MVC	Model View Controller
ORM	Object Relational Mapping
OOP	Object Oriented Programming
OS	Operating System
PC	Personal Computer
PDGA	Professional Disc Golf Association
PRO	Professional

SQL	Structured Query Language
UI	User Interface
UX	User Experience

Table of Contents

List of Figures

1. Introduction

Touring all around the world is a common scenario for an increasingly bigger number of players as disc golf keeps growing in a rapid pace. Every year, names from Finland and Estonia can be seen competing at the highest level in the USA and vice versa. These professionals have a strict schedule to stick to and they plan it as early as possible during the winter. Keeping track of registration dates and event dates to avoid collisions in the schedule is an extremely delicate topic. The schedule usually is created to make sense in the geographical aspect as well, like west coast touring is at the start of the year and halfway through the tour moves to east coast, but not inbetween every two weeks with players having to fly out all the time.

At a lower level, considering local professionals or even amateurs, while not having as strict of a schedule, they still need to plan their season, and currently, players have to navigate different web sites (covered also in analysis as part of this thesis work) and to combine information from different sites on paper or keep information in Excel worksheets or elsewhere in an uncomfortable manner. A player might have multiple databases for this purpose.

The purpose of this thesis is to create a web site that allows competing players in disc golf to plan their personal schedule, keep track of new events, if necessary create events for themselves and to combine the scheduled events with traveling data. Upon this a recommendation system is built to suggest events based on event tier, class, location and time. Purpose of this thesis also includes not having to manually write down dates and times and set alarms by having the possibility to export registration and event dates and times to their preferred calendar app. Built in is a visualization tool based on Google Maps API and scheduled events' location details. This thesis solves the no single source of truth problem for all events requirement, as players have event information, links to registration and abstract event info, travel information, date and time of registration info all in one place. This thesis helps players keep track of their goals by spreading the season into periods and enabling players to effectively fill out the schedule according to their goals.

Personal motivation behind the idea was repetitive manual handling of registration dates for events, setting notifications to get aboard events, having to separately store travel information and event info and every time when needing to check something it required logging in to multiple applications. There was no easy way to add events to calendar, especially cross platform (iOS, Android, Outlook on Windows etc) calendar files export capability seemed tempting. Motivating was also the possibility to visualize schedule on a map, which could be printed out and used as a canvas to draw arrows and events happening order (traveling plan) on. Sometimes players miss out on relevant events and for that a recommendation system was taken into project scope, first iteration, but already potentially very useful as it helps with events based on location, tiers and classification. The usual sites keep track of registered to events in schedule, but not simply interesting events (or own a separate schedule for followed events). Purpose was to combine interesting events in one place and not be dependent on registering status and give more freedom to players, along with custom events. The click-race to register to an event can not be avoided, so why not let player's favourite calendar app take care of the notification purposes, which it is built for? This project was designed from a personal requirement viewpoint, adapted to market needs and made to fit between existing applications.

1.1 Disc golf

1.1.1 What is disc golf and how is it played?

Disc golf is a game of getting the disc in the basket with the least throws possible. A disc golf disc is a round object, usually around 21 cm in width [5], a rim to hold on to and a flight plate which makes up the center of the disc – this is where the wind provides aerodynamic lift.

There are different types of discs, distance drivers, midranges, putters with their own unique flight characteristics.

Courses usually contain 9-27 holes, with 18 being the golden standard for a competition type event. This means that around 72-90 people can fit onto the course at a time in groups called pools of around 4-5 people. [6]

Every hole begins with a teepad to throw off of and a basket to be reached. The baskets are usually standardized. Game setup is somewhat similar to golf, first developed by hipsters in the USA [8] in the 1960s-1970s and in the early stages, played with a frisbee. A disc golf disc is much more wind resistant than a frisbee, enabling it to be controlled even from around 150 meters in the right hands.

Players who are not just simple Sunday throwers usually have an urge to compete with each other and especially amongst friends. This also has a socializing aspect – being the reason behind why competitions were first started - and there are all types and sizes of competitions.

Disc golf is all about trying to achieve the perfect form whilst being in naturally gorgeous parks and forests combined with the social aspect and being active outdoors.

Now that disc golf has expanded all over the world, from USA to Europe, Australia, Japan, Africa and so on, a need for centralized disc golf events planner has arisen. There are some implementations of this requirement and they are also covered in this document.

1.1.2 Types of events

There exist weekly unsanctioned competitions where you can get your local rating (i.e. in Europe *discgolfmetrix.com* is used and it is possible to get "*metrix"* rating). This is not the official rating in disc golf and neither does it affect PDGA.

PDGA events are distributed by size and minimum requirements into A, B and C categories, where A is the highest quality event. [7] There are also L type leagues spanning over a larger period of time, usually taking place weekly and you can get PDGA rating, being the official rating of professionals and amateurs alike.

Bigger events are described as NT (National Tour), Major events or Pro Tour events. [7] Here the highest rated players of the world compete in and usually these tournaments have to be supportive of all the players around the world (tournament organization in English, housing possibilities nearby, course quality and complexity etc).

1.2 Project scope

This event planner is focused on the player's own tournament schedule with quick event information search, previews and overviews making the events easier to navigate between. Each event is tied to an external PDGA link, website link and registration link, if present in the PDGA database, combining different sources and making it easy to navigate to them.

This site is not solely dependent on what events are added to a certain page but gives player also the freedom to create custom events for itself and adding events to schedule is not registration-based meaning the player has easy access to exact events it wants, especially through custom created events. This is in its essence a non-official planner by design, enabling players to add any kind of information desired to any event in the schedule, including travel information, costs estimation, registration dates, accommodation details and so on.

The planner builds a user habits based profile for every user, calculated upon existing events in the user's schedule, taking into consideration locations, time spans between competitions, event tier and other classifiers tied to the events. Upon this a recommendation system is built to ultimately provide the user with actually useful events data before the user even thinks of searching for it.

The planner helps combine events in Europe and USA by having users be capable of creating their own events combined with the search from PDGA database. For this, integrations with PDGA is required.

Planner keeps user data safe and sessions validated by Azure AD B2C, where web application and web API are registered and communication is securely validated.

Planner aids the players to reach their goals by providing a way of keeping players' desired number of tournaments in mind. The players set the wished amount and time of the year (i.e. March to October) and the planner calculates the average timespan between events, recommends some events based on location and helps keep the player on track.

Registration dates and event dates can be exported to the calendar as a .ics file to make player's life easier by not having to mark notifications manually every time. The calendar event is saved by the event name. Player notification is then handled by the specific calendar app used by the player.

The website also features visualization of events on a map, displaying the full schedule on it. An integration with Google Maps API was made for this cause.

2. Project description

The project started out with meetings every Wednesday with the supervisor. The first step was to specify the scope of the project in place. It took a little bit of time to find the real niche of the application, to make it stand out amongst other sites brought out in the market analysis.

Once the niche was found, development could begin by putting the project backbone in place and designing general architecture, including frontend, backend, ORM, migrations, database and register and login capabilities including connections between all different parts. This documentation is described in detail in the next chapter.

For project management TalTech Gitlab was used, a repository with the name „iaib“ was created as a monorepo containing both backend and frontend code. This is the version control system of the code developed as part of the thesis work.

In Gitlab, boards were used to track issues in 3 statuses: „To do“, „In progress“ and „Done“. Issues were tied with estimations and time spent. Following good principles issues were created mostly under 8 hours long except for some time lengthier Azure AD B2C integrations etc.

The application consists of a single page application (SPA) and a web API and a database to support all the functionality.

2.1 Use cases

The target audience consists of anybody who is planning their career in disc golf or looking to take it as a serious side hobby on a local professional level:

- Pro players playing tour stops in the USA, Europe and other regions, competitions of mostly A tiers and higher.
- Local players looking to tour in their home country and neighbouring countries, competition level A-C mostly, i.e Euro Pro Tour in Estonia, Sweden, Finland etc.
- Upcoming players (in Finland kids playing disc golf is in rapid growth thanks to schools and disc golf overall is pretty much becoming a national sport there [11]), for whom their parents help create and organize schedules so that all information is in one place.

For all of these groups some important use cases include:

- Quick registration. It is difficult to keep track of all dates and times of events for when the player can use its rating to register. For that, calendar exports are enabled once the player specifies the registration date suiting its rating. This also includes being able to export event dates helping prevent overlapping.
- Planning traveling, estimating costs, planning accommodation and car rentals. Now the player can store everything about the event in a specialized container meant for disc golf events. No matter if the event is from the PDGA or self-created.
- Distributing events all over the tournament year. Player can easily verify event overlap in the chain view of schedule to avoid scheduling collisions.
- Quick and up to date information about events. Event information is stored in the official PDGA events database and is never stored as per request from PDGA in their API usage agreement.
- Setting goals for the tournament year and helping players to reach their goals by simple dashboard metrics and some basic calculation of average required time between events. Events that are in schedule but timewise in the past are considered to be completed by the player.
- The player needs recommendations on upcoming events that might go unnoticed otherwise. This is based on the user profile of the player calculated from events data in schedule. Holes in schedule are taken into account and the planner tries to recommend some events into those slots.
- Visualization of the schedule on a map for printout and drawing purposes, exact marker pinpoints are automatically created from event coordinates.

2.2 Existing solutions

2.2.1 PDGA – www.pdga.com

Centralized official disc golf association site. Owner of the professional and amateur players database, source of rules, management of PDGA events, PDGA rating calculation, course information, media etc.

Being a user is contract based (can choose whether you are AM or PRO with the latter being able to cash every tournament’s prize). This site is also used to mark official competition scores (usually duplicated in Europe with discgolfmetrix.com and in USA, the UDisc).

Owner of PDGA API, enabling queries about events, courses, players as documented on their site against their database.

This site is not a direct competitor, as it is simply the source of official data and lacks the personal touch and functionality of this thesis project.

2.2.2 Discgolfmetrix – www.discgolfmetrix.com

A website made by a solo Estonian developer used for course management in Europe, events organization and registration and for keeping scores during tournaments. Includes separate registration phases for tournaments, owns separate rating called the *metrix rating* which is unrelated to the globally official PDGA rating.

Metrix rating is used for weeklies and smaller non PDGA events to separate players into appropriate categories based on skill level.

In Discgolfmetrix one can follow an event, register (if the appropriate phase is open for registration and you are quick). Displays upcoming events and you can check out your training rounds. You can also export the schedule in JSON format, potentially being an option to expand this project in the future by an integration.

This site is a competitor in the sense that people can register and keep track of their schedule, strength being that players can visualize their statistics of previous competitions, but lacks goal tracking or event recommendation functionality (only based on location) or good overview of schedule since it tries to do many more things for Europeans. No calendar exports possible either.

2.2.3 UDisc – www.udisc.com

USA’s version of Discgolfmetrix containing a lot of information about different courses and leagues, including a blog and a live metric system for checking scores called UDiscLive. Mainly USA’s events are organized in UDisc with rating system and scorekeeping.

In Europe, UDisc is not really widely used.

Strengths are the UDiscLive site, media side and score keeping and course info. This goes past thesis project’s scope and so is not a main competitor, as with Discgolfmetrix.

2.2.4 Disc Golf Scene – www.discgolfscene.com

Web application where you can check tournament information and register competitions. Can look up leagues, courses, clubs, tournament series, videos. Shows exact registration information (connected to PDGA), enables you to follow a tournament and check registered players information.

In Disc Golf Scene site you can also see your schedule, upcoming and past events, location of tournaments and cost of divisions. Under events, tournament series are also brought out and can be seen in a separate view. Overall a bit aged design in desktop view.

This is probably the main site to compete with. Strengths are that events are tied to event series, easy to find another event from the series and series is displayed on a map (but not your own schedule). It supports full registration as deeper integration with PDGA has been made here. This is not possible with using the public API. However, Disc Golf Scene site and registration is in this thesis' planner one click away from under event info, if registration is open. Players can visualize their own schedule and export to calendar important dates which one still can not do on Disc Golf Scene site. Disc Golf Scene lacks personal flair, but has much information considering the forum, media and more local info. Does not allow you to create custom events for yourself.

2.2.5 Other sports' sites

By searching for planners one is flooded with apps to create groups for events or similar groups management functionality. Planners usually are games about management or event planning tips. So the usual keywords fetch nothing interesting.

In golf and tennis more specifically, official sites exist, but these are overflowing with information and are not as focused on player itself. So a simple planner of this scale is pretty new since every site is trying hand in media, live scores and more. This functionality was left to other sites already in the project analysis.

To go into production, more analysis with players who compete in different sports (weight lifting, tennis, golf etc) could be made to learn, what they use and what is missing. Perhaps abstraction based on sport could be made so different sports competitors could use same core functionality, but with different sports.

3. Project design

Technological stack was greatly inspired by the full Microsoft stack, meaning ASP.NET Core, .NET Core, EF Core frameworks and SQL SERVER were used. C# is the strongly typed object oriented programming language used here. All of these components are built to be easily integratable with each other and there are multiple tutorials of this on Microsoft's official documentation site. ASP.NET CORE app itself in this project serves the role of a web API and UI is handled by one of the most popular Javascript frameworks called React, combined with state management library Redux. On the styling side, some vanilla CSS was combined with Bootstrap v5 classes and components. This is the newest version and can save a developer some time as a CSS library.

Microsoft Azure services were used to handle authentication and data of users, more specifically, Azure Active Directory Business to Consumer (AD B2C). The Microsoft stack was familiar to me and since its documentation base is huge and it is very capable and well optimized and modern, it was taken into use. Redux and Azure AD integration were fully new for me, but also popular and capable choices.

3.1 Backend

For this project C# and the full .NET stack by Microsoft is being used. .NET Core and ASP.NET Core and EF Core are the recommended frameworks for new projects by Microsoft, by far the largest software company out there. [1] Everything is on version 6, the newest non-beta version. [9]

Microsoft developed C# in 2000. Its architecture adops the best features of Java and C++ and it was really easy to obtain after Java, as both are strongly typed OOP languages. One of the main use cases for C# is web application development and there is huge amounts of documentation from Microsoft and it was really nice to use. Except in some parts where information was on contrary to other pages, but both were filling their own specific requirement so learning to properly navigate in the documentation took some time.

Backend architecture was inspired by Domain Driven Design principles (DDD). [10] There are two basic components, WebApi and Core. WebApi is the Api provider, app startup configurer and controller base designed for external usage (by Web App). ASP.NET Core v6 was used for WebApi, which is the most up to date version of ASP.NET and Core meaning it is runnable on every OS.

ASP.NET can be used for Web Apps (called ASP.NET Core MVC) as well as Web APIs. Since JS applications are much more common and more widely supported, it is only used for Web API purposes in this project.

Core project uses .NET Core version 6 and is the basis for the application business logic, models, database connection, service base. Core is also responsible for external services meaning mainly integration with PDGA, login, token handling and search queries.

Core project is tied with EF Core, which is an ORM meaning it tracks certain class models and creates database tables out of them, each table having its own configuration and context registration under base context. Through EF Core database migrations are also configured meaning application owns a version control system for database. Each new version saves a snapshot of previous design and applies new changes upon it. For this, Add Migration and Update Database commands are required. It is very easy to drop or create a database from 0 just by having the model information in the migrations.

3.2 Database

The database is based on predefined and preconfigured class models in the Core project, tracked by EF Core and turned into migrations. Then a connection with SQL Server is made and a database is created and migrations are translated into SQL.

Database language is Microsoft SQL and database management system used is SQL Server Management System by Microsoft.

Migrations and through them, database version is tracked in the __EFMigrationsHistory table (default). There are lone standing SystemOptions and SystemTokens tables for environment specific configuration and PDGA integration specific activities respectively. (Figure 1)

Schedule is the master of saved external events and events. On the loading of event properties, schedule id is used to gather the necessary properties and then it is possible to divide them amongst the different types of events.

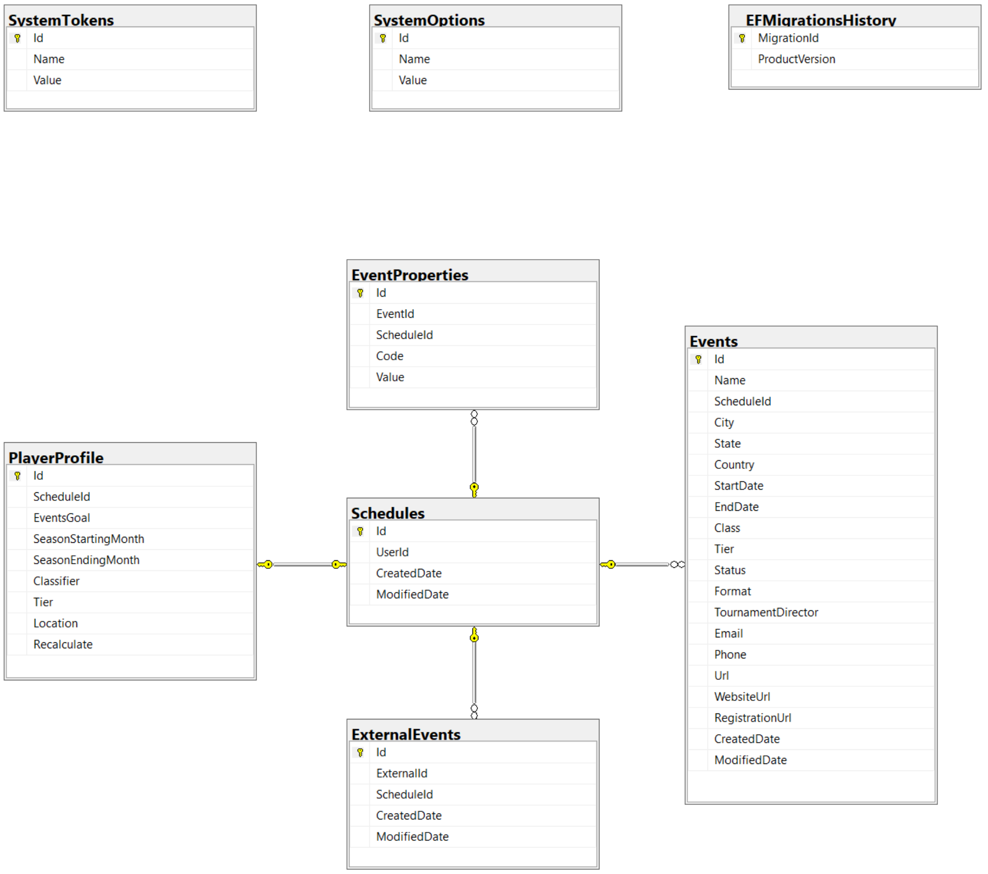


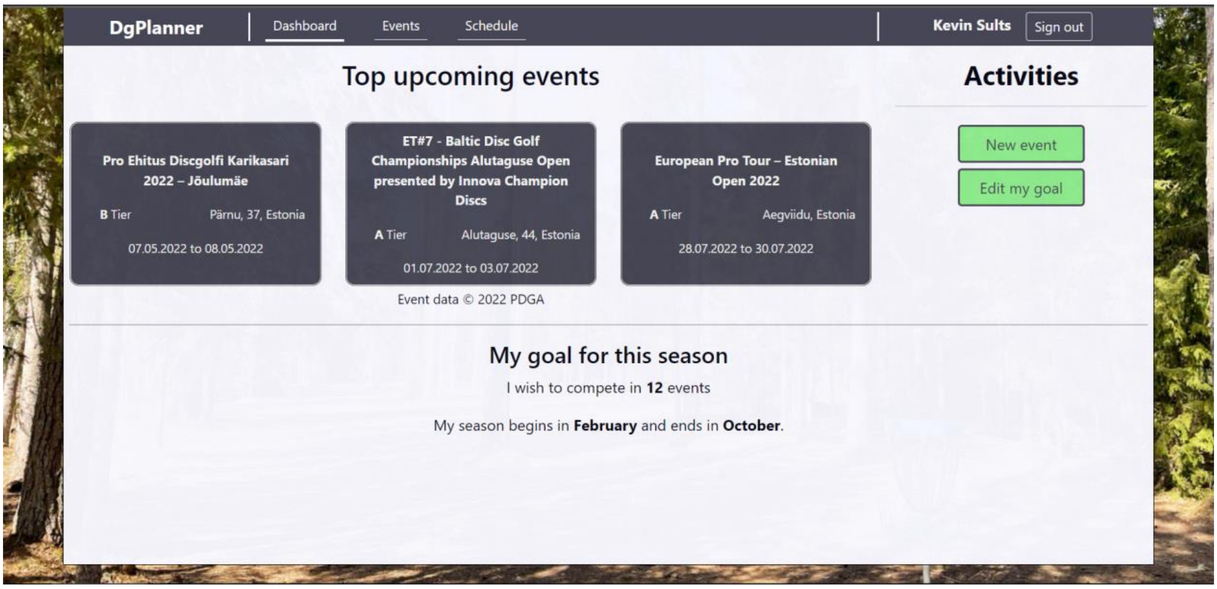
Figure 1 Database schema

3.3 Frontend

React was chosen for the frontend technology as it is currently one of if not the most popular JS framework. React was created by Facebook and is meant to be used to build highly responsive and functional user interfaces. Combined with some state management by Redux, again a well documented framework with active community, it enables easy creation of websites while keeping the learning curve rather small. [2], [3], [4]

Styling is a combination of CSS with Bootstrap v5, more specifically React Bootstrap. This provides the developer with multiple premade classes and components for convenient use.

User experience takes into account what is important for the user – events, quick previews with clear information about events, overview of schedule and personal data tied to events and visualization of full schedule on map. On the dashboard (Figure 2), it is made clear what are the upcoming events and possible actions in the application. Players can create new events and edit and view their goal. Events are being fetched from schedule, ordered by starting date in the descending order.



Top upcoming events

Pro Ehitus Discgolfi Karikasari 2022 – Jõulumäe
B Tier Pärnu, 37, Estonia
07.05.2022 to 08.05.2022

ET#7 - Baltic Disc Golf Championships Alutaguse Open presented by Innova Champion Discs
A Tier Alutaguse, 44, Estonia
01.07.2022 to 03.07.2022

European Pro Tour – Estonian Open 2022
A Tier Aegviidu, Estonia
28.07.2022 to 30.07.2022

Event data © 2022 PDGA

Activities

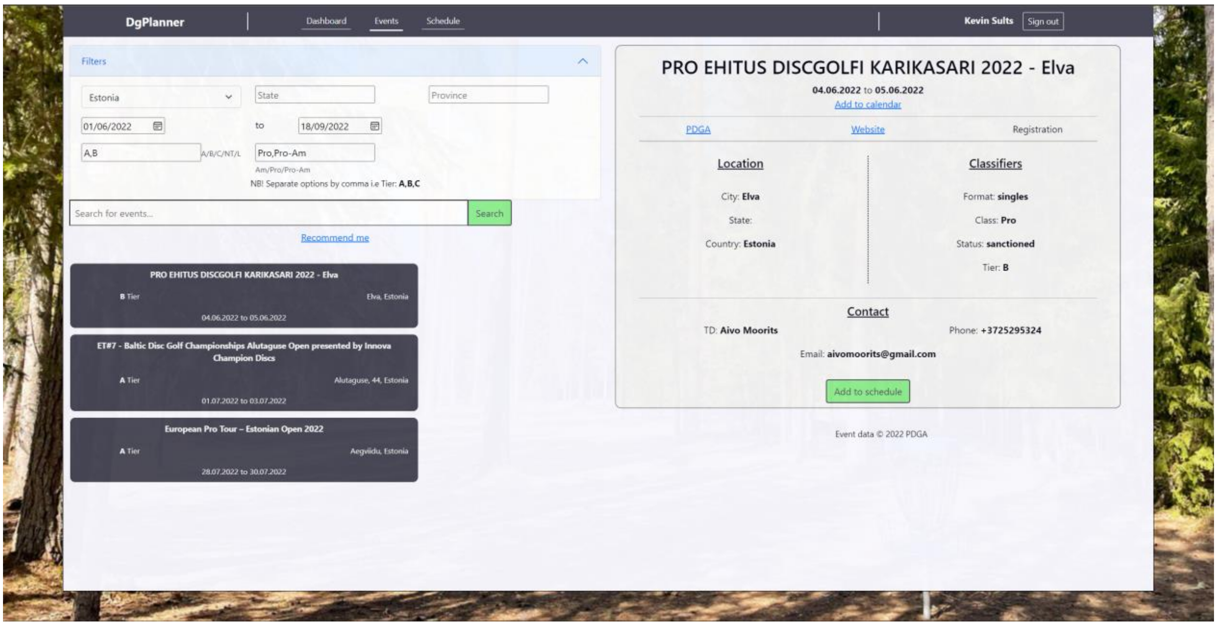
My goal for this season

I wish to compete in **12** events

My season begins in **February** and ends in **October**.

Figure 2 Dashboard view

In the events tab of the project (Figure 3), players can search for events by name, or by specifying any of the advanced filters values. Hints and tips are given in the UI for smoother user experience. Clicking on an event provides the user with a full view of the event's details. User has here the possibility to prefill search, this recommendation is based on events in user’s schedule, upon which a profile has been created.



Estonia
State
Province
01/06/2022 to 18/09/2022
A,B A/B/C/NT/L
Pro,Pro-Am
Am/Pro/Pro-Am
NB! Separate options by comma i.e Tier: **A,B,C**
Search for events...

PRO EHITUS DISCGOLFI KARIKASARI 2022 - Elva
B Tier Elva, Estonia
04.06.2022 to 05.06.2022

ET#7 - Baltic Disc Golf Championships Alutaguse Open presented by Innova Champion Discs
A Tier Alutaguse, 44, Estonia
01.07.2022 to 03.07.2022

European Pro Tour – Estonian Open 2022
A Tier Aegviidu, Estonia
28.07.2022 to 30.07.2022

PRO EHITUS DISCGOLFI KARIKASARI 2022 - Elva
04.06.2022 to **05.06.2022**

Location
City: **Elva**
State:
Country: **Estonia**

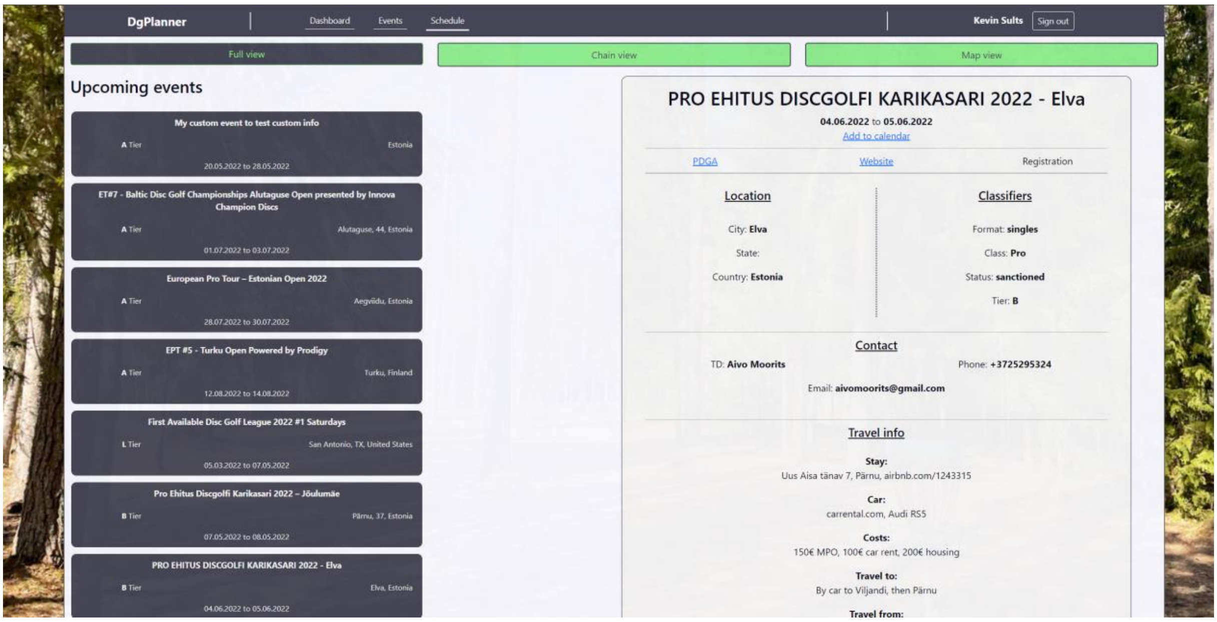
Classifiers
Format: **singles**
Class: **Pro**
Status: **sanctioned**
Tier: **B**

Contact
TD: **Aivo Moorits**
Phone: **+3725295324**
Email: **aivomoorits@gmail.com**

Event data © 2022 PDGA

Figure 3 Events view

Schedule tab view (Figure 4) shows players their events, which are added to schedule, ordered by starting date in descending order, with past events being shown at the bottom. Clicking an event allows the player to remove events from the schedule or edit event data. In case of a custom event, all data is editable, in case of PDGA event, only personal traveling info is customizable. Here the player is able to export calendar files of event (shown under event date) and of registration date (if specified) (Figure 5). Registration date has the capability of specifying a time as it is essential to quick reaction time.



Upcoming events

My custom event to test custom info
A Tier Estonia
20.05.2022 to 28.05.2022

ET#7 - Baltic Disc Golf Championships Alutaguse Open presented by Innova Champion Discs
A Tier Alutaguse, 44, Estonia
01.07.2022 to 03.07.2022

European Pro Tour – Estonian Open 2022
A Tier Aegviidu, Estonia
28.07.2022 to 30.07.2022

EPT #5 - Turku Open Powered by Prodigy
A Tier Turku, Finland
12.08.2022 to 14.08.2022

First Available Disc Golf League 2022 #1 Saturdays
L Tier San Antonio, TX, United States
05.03.2022 to 07.05.2022

Pro Ehitus Discgolfi Karikasari 2022 – Jõulumäe
B Tier Pärnu, 37, Estonia
07.05.2022 to 08.05.2022

PRO EHITUS DISCGOLFI KARIKASARI 2022 - Elva
B Tier Elva, Estonia
04.06.2022 to 05.06.2022

PRO EHITUS DISCGOLFI KARIKASARI 2022 - Elva
04.06.2022 to 05.06.2022
Add to calendar

PDGA Website Registration

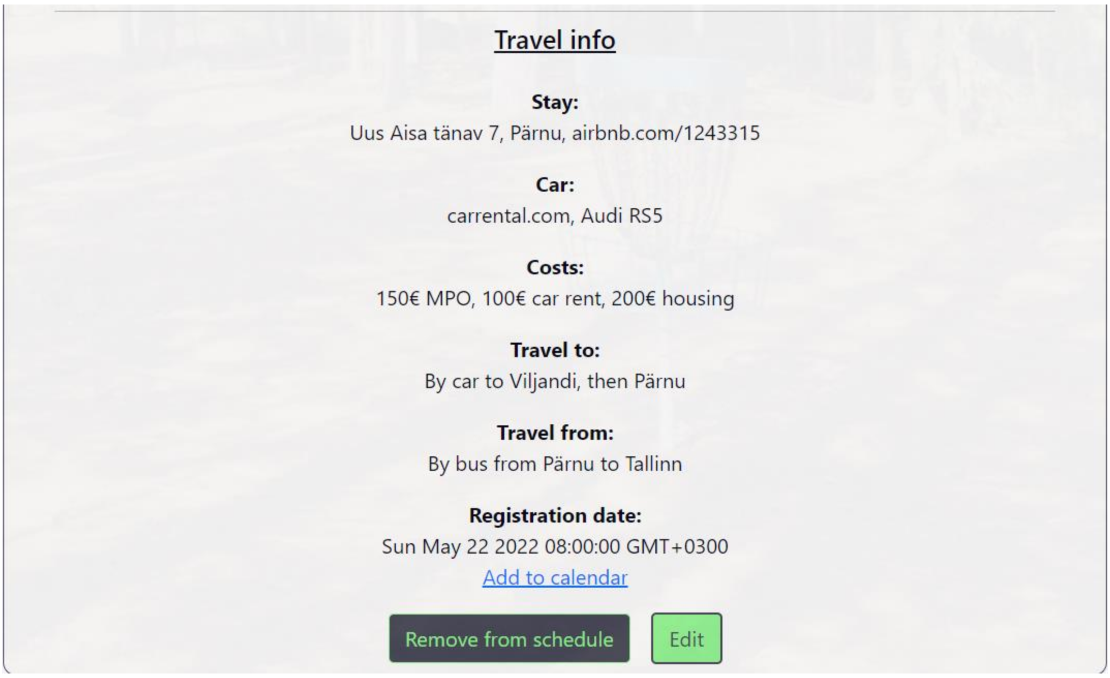
Location
City: Elva
State:
Country: Estonia

Classifiers
Format: singles
Class: Pro
Status: sanctioned
Tier: B

Contact
TD: Aivo Moorits Phone: +3725295324
Email: aivomoorits@gmail.com

Travel info
Stay:
Uus Aisa tänav 7, Pärnu, airbnb.com/1243315
Car:
carrental.com, Audi RS5
Costs:
150€ MPO, 100€ car rent, 200€ housing
Travel to:
By car to Viljandi, then Pärnu
Travel from:

Figure 4 Schedule view



Travel info

Stay:
Uus Aisa tänav 7, Pärnu, airbnb.com/1243315

Car:
carrental.com, Audi RS5

Costs:
150€ MPO, 100€ car rent, 200€ housing

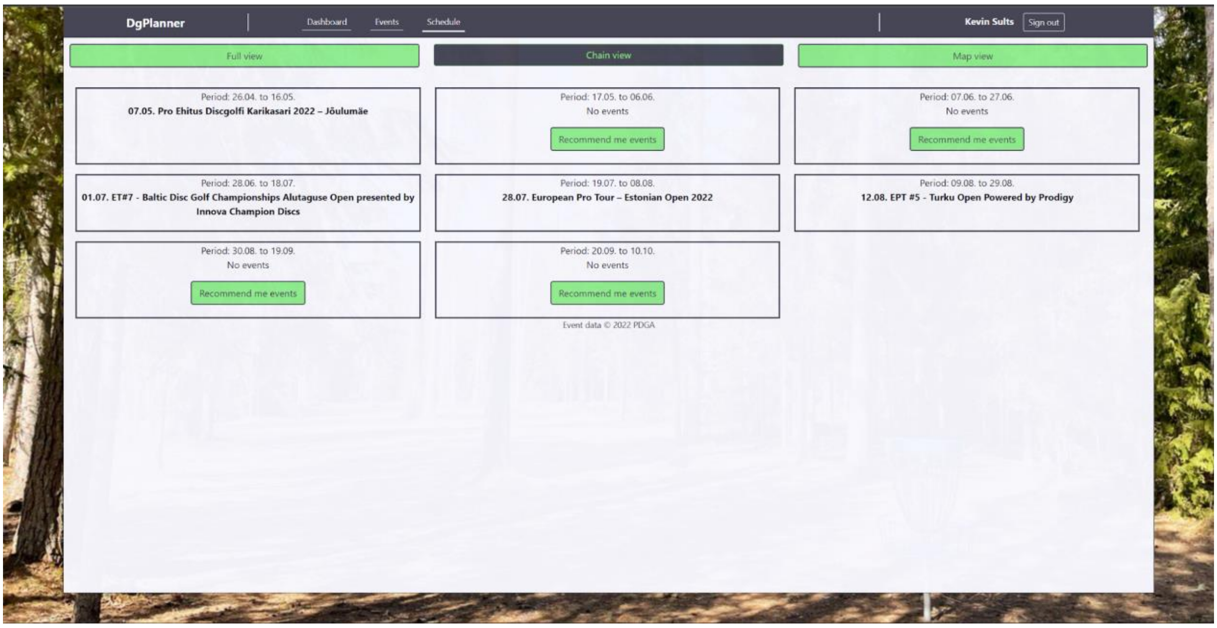
Travel to:
By car to Viljandi, then Pärnu

Travel from:
By bus from Pärnu to Tallinn

Registration date:
Sun May 22 2022 08:00:00 GMT+0300
Add to calendar

Figure 5 Full travel info and action buttons of an event in schedule

The chain view of the schedule provides player with a fuller understanding of forecoming season. Season takes into account season starting and ending month set in the player goal on the dashboard (Figure 2). All past events are considered as completed events. Remaining season is divided into periods called chain links and every period has its own events happening during that period (events that are in the schedule) and its own recommendations. Player is enabled to quickly search for an event in an interesting time frame shown as the link by clicking “Recommend me events” button. This initiates a few actions - player is navigated to events view (Figure 3), advanced search filters are prefilled to fit the link recommendations and timespan and an automatic search is triggered for the player. As events are searched from the PDGA, player can quickly specify its needs for the timespan or narrow the search down. This is the basis for a filled schedule, filled being measured by the amount of events set in the player goal compared to amount of non league type events in schedule.



Period: 26.04. to 16.05.
07.05. Pro Ehitus Discgolfi Karikasari 2022 – Jõulumäe

Period: 17.05. to 06.06.
No events

Period: 07.06. to 27.06.
No events

Period: 28.06. to 18.07.
01.07. ET#7 - Baltic Disc Golf Championships Alutaguse Open presented by Innova Champion Discs

Period: 19.07. to 08.08.
28.07. European Pro Tour – Estonian Open 2022

Period: 09.08. to 29.08.
12.08. EPT #5 - Turku Open Powered by Prodigy

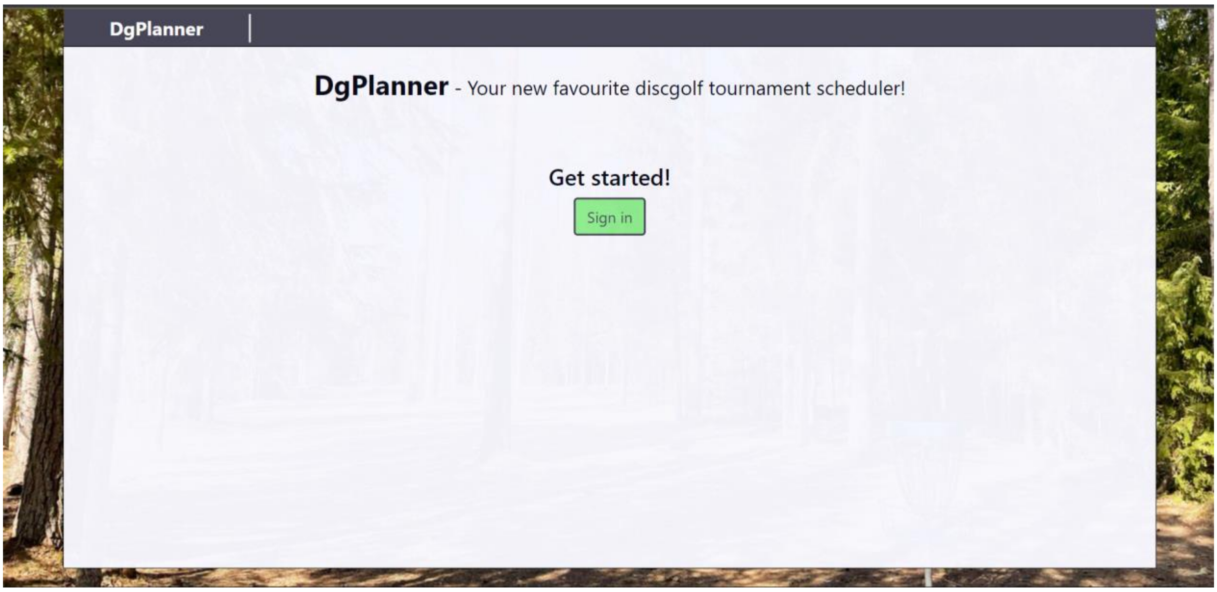
Period: 30.08. to 19.09.
No events

Period: 20.09. to 10.10.
No events

Event data © 2022 PDGA

Figure 6 Schedule in chain view

Figure 7 shows a simple welcoming page with menu links removed, and as no functionality is supposed to be used without the connection of an user, not much is being shown in this page. This is just enough to get a player started on their journey to planning a full schedule for the year.



DgPlanner

DgPlanner - Your new favourite discgolf tournament scheduler!

Get started!

Figure 7 Welcome page

3.4 Integration with Azure AD

From the talks with my supervisor one thing came out that was not planned in the initial project scope - to include OAuth 2 for login purposes. That is why Azure AD B2C support was added to the project, enabling future expansion to social media logins.

In the beginning, integration with Azure AD was implemented, but that was a wrong choice for the application needs since the B2C version was the right version for social media and personal account logins. A migration to B2C was then made.

There is a specific login and registration flow in Azure AD where all this could be configured (Figure 9), i.e. what data application requires from the user, where are the endpoints to turn to for session tokens and where to validate tokens from backend side. All this is customized for this application and design can further be changed to look as if the login page is hosted by the application itself not in Microsoft Azure to create a smoother login flow for users. (Figure 8)

For now, there exists only single tenant options for users meaning users have to register with a password for this application, but multi-tenancy is possible in the future, as supported by the AD B2C.

In Azure AD one has to register applications as shown in Figure 10, most commonly a Web App and a Web API to serve that app. From Web API side, API scopes can be exposed to be used in the Web App, which will later be used for workflows, all existing in token requests sent to the Microsoft authentication server.

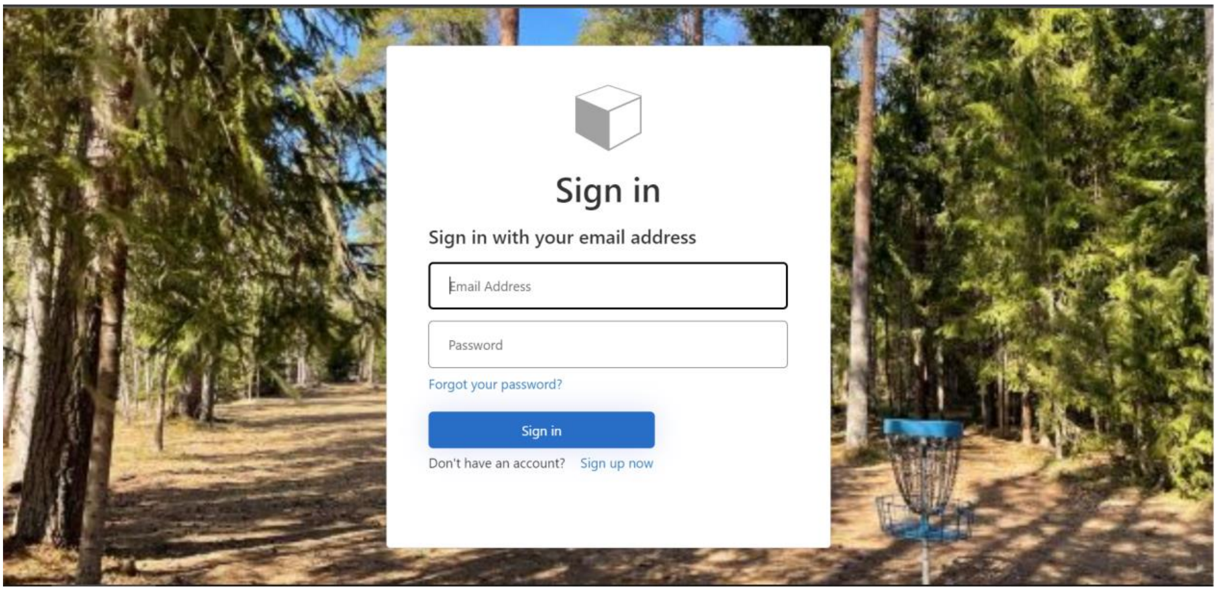
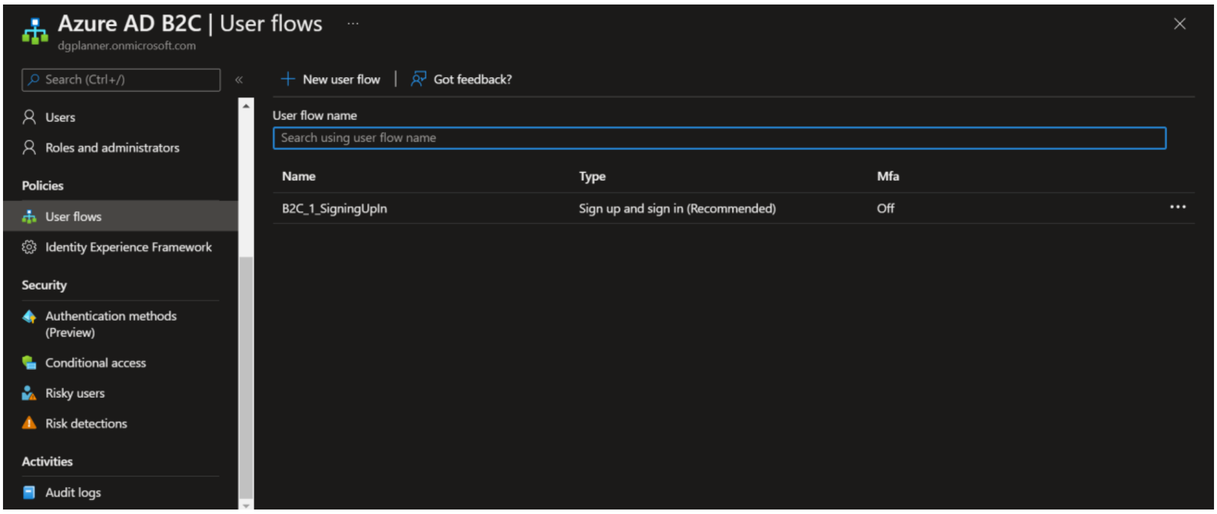
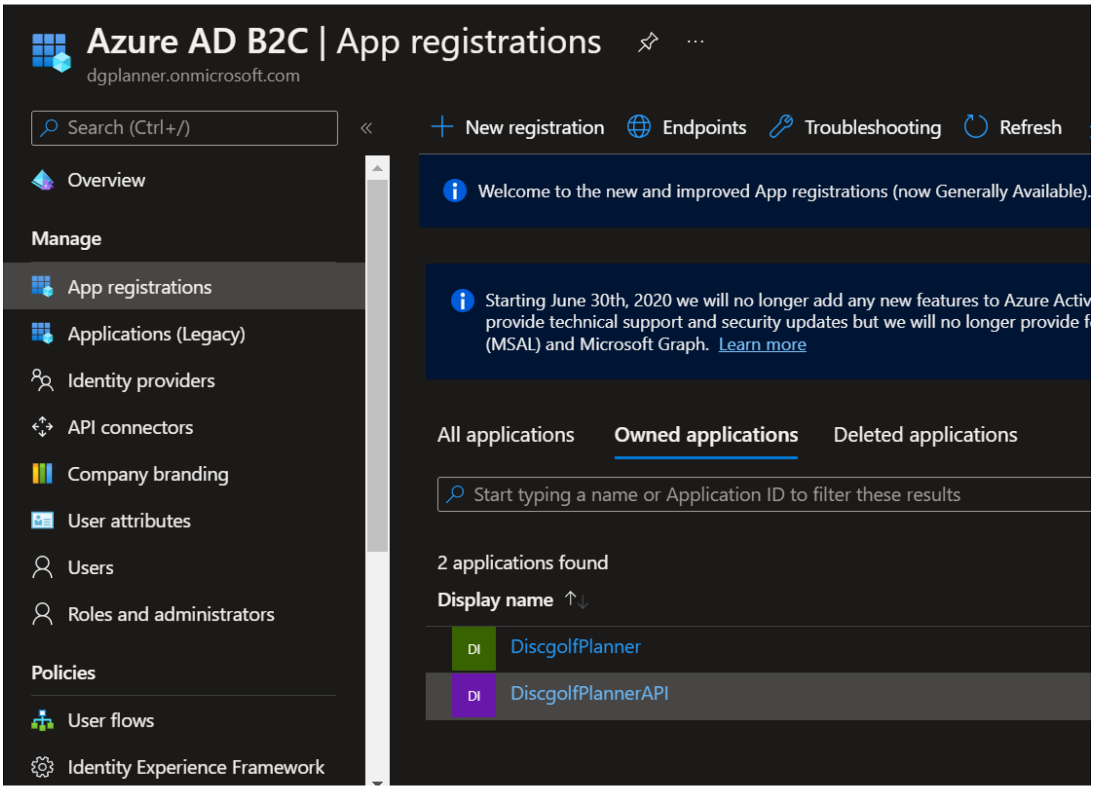

Sign in
Sign in with your email address
Email Address
Password
Forgot your password?
Sign in
Don't have an account? Sign up now

Figure 8 Sign in page hosted by Azure AD B2C



Name	Type	Mfa
B2C_1_SigningUpIn	Sign up and sign in (Recommended)	Off

Figure 9 User sign up and sign in flow registration in Azure AD B2C



Welcome to the new and improved App registrations (now Generally Available).

Starting June 30th, 2020 we will no longer add any new features to Azure Activ
provide technical support and security updates but we will no longer provide f
(MSAL) and Microsoft Graph. Learn more

2 applications found

Display name
DiscgolfPlanner
DiscgolfPlannerAPI

Figure 10 App registrations in Azure AD B2C

3.5 External services

In order to display events from the official site, an integration with PDGA API had to be made. For that, one has to be a user in PDGA (own an AM or PRO contract) to register as a developer user. Access is then granted and account initials are saved in the database securely, not in code.

If tokens are invalidated in response, a re-login will be made. On login, session token name and id are saved into SystemTokens table and queried on every event search made by the user. All this enables minimal queries to PDGA and offers the smoothest experience for the user.

PDGA enables search by very specific parameters and has the response documented in JSON format on their website. Integration process went smoothly, however on some days every query to the PDGA API takes about 40+ seconds. This means it is not very usable on a daily basis and they have documented the API to be “as is”. Event search parameters are also quite different from what was expected and an adaptation had to be made and as a result, now the user sees clear search parameter recommendations in the UI.

For calendar exports, React iCalendar Link node package was used. Integration was straightforward and support for event and event registration was added.

To visualize player schedule, integration with Google Maps API had to be made. (Figure 11) This included registering as a developer in Google Cloud Platform. Here a 300$ amount of free credit was instantly added to my account to use for Google API services.

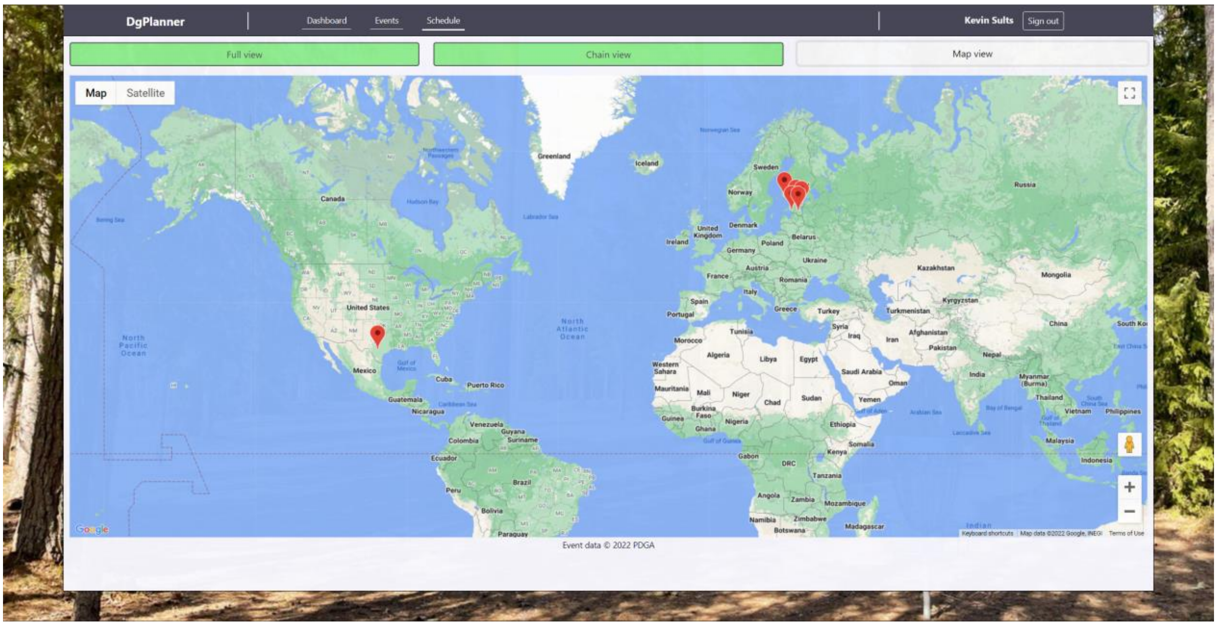


Figure 11 Visualization of schedule on integrated Google Maps

4. Results and validation

4.1 Results

As a result of this thesis, a fully functional website was developed. It is meant to support players touring in the USA and Europe or even all over the world. This does not exclude local players. The application has a supporting web API and integrations with Azure AD B2C as user and session token base and PDGA, which is the source of truth for official up-to-date information and Google Maps API.

Roles of the application include helping players plan their year in competitions, helping them keep track of their desired goals and enabling players to save traveling and personal event data along with the events. Keeping track of player’s season goals includes recommendations for slots in the schedule where potential events could be fit. A search is filled out for the user by the system, depending on which slot interests the user most, and fresh event data from PDGA is then served in the web application UI.

Player has the ability to build a schedule up, adding events and binding events with travel information. Integration with a calendar is made easy – export event dates by clicking „Add to calendar“. Bind an event with a registration date and time and player can also export that. Schedule is visualized in multiple views, full view, chain view and map view, made with Google Maps integration. Map view of schedule helps players visualize their goals on a map by showing their full schedule. Chain view keeps players' schedules organized and spread out evenly during the season.

This application is built to be a single source of truth for events. Every single event is tied to separate official links from PDGA and official partner links.

4.2 Validation

There was a chosen use case to play through given to every validator (usually a player ranging from amateur to local pro level). This use case was the same for everyone and after completing the tasks, a Google Forms form had to be filled out. Validation results are based upon this.

4.2.1 Chosen use case

Player has to navigate to website page (hosted in my computer), register an account and log in. Registering is by email and password. Then a player starts off by setting its goal for the season, followed by searching for a couple of events, first by name, then by making use of the advanced search and then simply by location. This familiarizes player with the search UI.

Now it is time for player to check its schedule and once it is confirmed that all the added events are gathered, time to create a custom event by using the „New event" button on dashboard. Players have to figure out navigation by themselves. Now the player has to check that all the information about the event was saved correctly and move on to add travel information to one official PDGA event and save.

This is where the player is tasked to go to „Chain view" of the schedule, to get a good overview of current schedule and try the recommendation system to fill a slot in schedule. Now player is tasked to search for the number of registered players under an official event. For this, navigation to PDGA page is required from event detail view, present under every PDGA event. Player has to export this event as a calendar file for itself and to check the contents. Continuing calendar integration, player is asked to fill the registration date of the event and to export that into calendar as well.

Finally, the player has to visualize the built schedule on the Google Maps map. This is the end of the use case and player is directed to navigate to Google Forms page and fill out the feedback.

4.2.2 Feedback

After completing the use case, potential users had to fill out a Google Forms document asking about overall user experience of the app, short feature set analysis, application usefulness on a daily basis for user itself and whether the player feels like this application fills a unique enough spot on the disc golf web application market. In the end, there was a free text format answer about recommendations for the app, what could have been made differently and what would they like to see included if they were to start using it.

Short overview of the use case and feedback process was also asked. Answers were tied to the skill level of player, ranging from 1 (amateur) to 3 (professional). 2 players rated themselves amateurs, 1 a Pro-Am and 1 professional.

All in all there were 4 validators with different skill levels and requirements, as one is a local professional competing in nearby countries, 2 are actively competing players and 1 is more of a Sunday player trying to get better and to get out there in terms of competitions.

2 players had a good experience of using the app and 1 rated the app even great! 4th had a bit more critical mind focusing on the potential and put „Clunky and a bit uncomfortable“ for the experience. The feature department of the app got an even better result, as 3 rated the collection of features to be very interesting and 1 put „Fascinating set of features, cleverly combined“. This is a promising basis for the future.

Then the readiness of the application was being evaluatued, 2 people thought they needed to test it more, but maybe could already make some use of it for themselves, one put that „This application is in beta, but I want a full product. Will wait and see.“ and left under comments that mobile version was highly expected. 4th one had great enthusiasm and answered „If I could, I would already be using it“.

So as the feature base was good, readiness could be better, uniqueness was now being evaluated. 3 out of 4 players rated the site to be „A rather unique perspective for a disc golf site, separates itself from others“, last one answering „Equal site among others, especially considering short development time“. This validates the good work specifying the niche at the beginning of development with my supervisor.

Overall recommendations and comments for the app included these thoughts:

„The idea is great and unique, I cant think of any other app that presents this kind of function. I would like to see this site as an app in smartphones and get notifications about events even if I am not logged in (app is in the background). In the future, there might be some ads/news section about hottest disc golf news in Estonia or some links to web disc golf shops.“

My comments about this would be that interest for a mobile application is definitely there and with some design and comfort tweaks it would be a good experience for the users. The great loading times mentioned were caused by slow PDGA API, as documented before and expanded on in the upcoming Comments section.

„Amazing experience! Didn't know I needed this until I had the oppurtunity to try it out!“

- (translated) „Since I am an amateur and do not compete in bigger events, I could still make use of the planner by creating new events fitting the description of local weekly leagues. Or even to plan practice rounds with friends, sometimes spanning over multiple days and with full travel requirements, for this I really could use the travel info tracking alongside my custom events. Exciting application, but I see more usefulness in it for more professional type players, but I can also appreciate the opportunities for amateurs. Map view I really liked, but chain view was a bit too complex to understand. Maybe in search one could specify that the main box is an event name not event location search bar.“

My comments on this is that I did not realize the full potential for amateurs in the beginning, but it was pleasing to hear that players are already thinking of how they could make the application work for themselves specifically.

5. Discussion

5.1 Integrations with Azure AD, AD B2C

Azure AD has only multi-tenancy and work/school and local account support, however, this came out a bit too late as I had already put multiple hours into integration. Thankfully, in the Azure Portal, I could register under AD B2C version instead, which supports all the social media logins and local accounts, which was the exact requirement of this application.

From the application side, only some URLs had to be changed to connect to the user flow and enable API scopes through which backend and frontend communicate (two separate registrations in AD B2C). B2C itself is partly in beta right now and has not yet the full functionality of AD, but it was a must considering the project's business needs.

Overall, a very interesting, but time consuming experience enabling perhaps in the future to better make use of Azure’s potential.

5.2 Integration with PDGA API

There were definitely multiple surprises considering using the PDGA API. First off, location search is enabled by two character country codes not names, which was a setback. Later, when schedule was created, I discovered that I had to query events one by one through event ids and gather them all together and send them back to the frontend. On slow API days, this takes number of events * 42 seconds worth of time.

Aforementioned slow API day in PDGA context means that every query unexplainably takes around 42+ seconds to return with a response. And sometimes the API works as normal, a few seconds per request at most.This is a huge problem for production purposes.

PDGA also demands copyright disclaimer text on every page where their data is shown and saving the data is punishable by player ban from PDGA API (and perhaps PDGA site itself). This is why event ids had to be saved to get schedule view, this is combined with custom events by players, both are tied together, but custom events are kept in web API database itself and take little to no time to query.

One more thing, I had to sign up for the PDGA developer program through my personal PDGA account, which I use to compete on a biweekly basis. This is a very important subscription based account for my hobby career and now all of a sudden I might have to go live under this account. This is a bit of a scary thought and serious security implementations have to be put in place to disable any kind of spamming or abuse from users through my personal PDGA account. The account is used to get session tokens from PDGA through which queries can be made.

The API itself doesn’t have very many opportunities to search for events, with direct access to the database, like on the PDGA website, one could see how it is possible to comfortably build a search around it. However, pretty much the best that could be done was done during this thesis in this project.

5.3 Problems with publishing of the app

First off I tried renting a Windows server from Azure Portal as I have built this project on full Microsoft stack, but even the cheapest version cost around 70$ and since I could not get free credit, that was the end of it. There are usually options to get 200$ for new users for one year and 100$ for a student, but neither of them worked for me. I do see a student subscription from TalTech.

Second option was the hugely popular AWS by Amazon, renting a server cost also, except for the cheapest options. However, I immediately noticed the weak parameters of the resources of the virtual machine that I would be given. I still registered for a machine and even got access through Remote Desktop. First thing was to install Edge using the preinstalled Internet Explorer but as that operation already took me around 10 minutes to complete (it did not start downloading) I decided that this is no suitable environment for my app hosting.

Then I decided to host the app from my own PC. For this, I registered application pools and sites under IIS, provided by Microsoft, and got them running. Now I had to publish my applications to public net and for this, I opened a firewall port matching the port on which the application UI was running. However, no access could be made from my public IP address to my computer, making me suspect that the physical port of my router's could be closed. Logged into admin settings of the router and I could not find port settings anywhere. After an hour of searching for answers, I tried registering a virtual server matching the port and my PC's local address. Tried to gain access again, but nothing. After some more searching in the web, I started to suspect the use of CGNAT by my operator, meaning pretty much all the router's settings were rendered useless. At least this is what I read from the forum. I decided after this not to pursue anymore as time was of essence and had people test the application on my own PC. For production purposes, a proper cloud server would be rented anyways.

6. Summary

The project scope set out during the analysis was fully developed. Players have a place to keep their events data as a whole schedule consisting of PDGA and own events, with possibilities to visualize it into different views and export important dates and times to calendar. Integrations with the PDGA API, Google Maps API and Azure AD B2C were made. Players are also enabled to fill out their goals for the season and to get recommendations based on unfilled slots in the schedule. The events can be tied to personal information for a all in one place experience, with important web site links being one click away under event information.

This project could use more development to go into production, since its functionality is easily expandable and a more complex recommendation system could be built. A major issue with going into production would be the volatility of PDGA API – it is just not usable enough with 40+ second queries every other day. The effectiveness of the application minimizes this effect in the use of the application as minimal inquiries are made. Here the next step would be to communicate with the PDGA and maybe give some improvement ideas on their API, such as multiple event id searches at a time not one by one etc. Integration with Discgolfmetrix would be a good idea to automatically import player schedules.

With enough improvements, potential to reach thousands of aspiring players could take this application into use, however, it would have to remain a non-profit project as PDGA API requirements declare so. The requirements also foresee that on every page where PDGA event data is displayed, a small disclaimer is shown indicating the origin of the data, which is of course, the PDGA.

It was fascinating to work with Microsoft Azure portal and services and let myself be guided by the massive website that is Microsoft Docs, containing all the information one could ask for on a single site accompanied by extensive tutorials and examples. Azure AD B2C enables the support for social media logins, definitely being one of the first improvements I would make in the future.

Players who tested the app were intrigued by the set of features that really differentiate this application from all others. They showed interest in the app, as long as it is continously developed for mobile as well, along with some design and comfort improvements in UX department. This I heartily agree with and it was also brought out that the application has all the potential to be huge. One player wanted to see even more integrations in the app to centralize disc golf information even more, perhaps creating a media perspective for the app as well. It was clear that players started figuring out how they could make this app useful for themselves, whether it was just by combining a few more features, using some of the features in a different manner or just straight up start using the application. This application really opened the mind of the high level professional by having them feel the sudden need for something like this, thanks to the innovation that went into this app. Players and I were interrupted by the slow loading times of PDGA events with only one player being able to complete the use case while loading was fast (under a second compared to 42+ seconds every query). I could also see the immediate effect of a much better user experience rating later in the answers. For production purposes, talks with the PDGA have to be undergone to find out the solution here.

All in all the application mostly fulfilled the set out purpose in the beginning, unfortunately missing publication of the app for public web by not wanting to pay for server rent and not being able to open local network ports for outside and a mobile version for which I did not have time for. The player can manage their schedule with recommendations based on previous habits, recommendations being a basic version as it is the first iteration of the system. Players are able to visualize their schedule in periodical, map and full views, each event binded with traveling information combined with official event info. Players can set their goals and the application helps them keep track of it. All this creates a centralized feel to the web site and has changed the minds of a few players already.

Bibliography

1. **Reiff, Nathan.** Investopedia. *Ivestopedia Web Site.* [Võrgumaterjal] 07. August 2020. a. https://www.investopedia.com/articles/personal-finance/121714/worlds-top-10-software-companies.asp.

2. **Gathoni, Mary.** MUO. *MUO Web Site.* [Võrgumaterjal] 29. January 2022. a. https://www.makeuseof.com/most-popular-javascript-frameworks/#:~:text=1%20React%20JS.%20React%20surpassed%20jQuery%20as%20the,features%20of%20frontend%20frameworks%20like%20Angular%20and%20React..

3. **Arora, Simran K.** Hackr.Io. *Hackr.Io Web Site.* [Võrgumaterjal] 26. April 2022. a. https://hackr.io/blog/best-javascript-frameworks.

4. **Statistics & Data.** Statistics & Data. *Statistics & Data Web Site.* [Võrgumaterjal] 2022. a. https://statisticsanddata.org/data/the-most-popular-javascript-frameworks-2011-2021/.

5. **PDGA.** PDGA. *PDGA Web Site.* [Võrgumaterjal] 10. February 2021. a. https://www.pdga.com/technical-standards/manufacturer-guidelines.

6. —. PDGA. *PDGA Web Site.* [Võrgumaterjal] 8. October 2020. a. https://www.pdga.com/pdga-documents/tour-documents/pdga-course-selection-guidelines.

7. —. PDGA. [Võrgumaterjal] PDGA Web Site. https://www.pdga.com/td/events-guide.

8. —. PDGA. *PDGA Web Site.* [Võrgumaterjal] https://www.pdga.com/history.

9. **Microsoft.** Microsoft Docs. *Microsoft Docs Web Site.* [Võrgumaterjal] 2022. a. https://docs.microsoft.com/en-us/.

10. **GeeksForGeeks.** GeeksForGeeks. [Võrgumaterjal] GeeksForGeeks Web Site, 24. June 2020. a. https://www.geeksforgeeks.org/domain-driven-design-ddd/.

11. **frisbeegolfliitto.** frisbeegolfliitto. [Võrgumaterjal] frisbeegolfliitto Web Site. https://frisbeegolfliitto.fi/disc-golf-in-finland-statistics/.

Appendices

Appendice 1 - Non-exclusive licence for reproduction and publication of a graduation thesis[1]

I, Kevin Sults,

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis “Development of Disc Golf Event Planner“, supervised by Gert Kanter,

1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;

1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.

2. I am aware that the author also retains the rights specified in clause 1 of the nonexclusive licence

3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

29.05.2022

[1] The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the nonexclusive licence, the non-exclusive license shall not be valid for the period.