



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**WEBOVÁ APLIKACE PRO OVLÁDÁNÍ
CHYTRÉHO SKLENÍKU S PRVKY GAMIFIKACE**

WEB APPLICATION FOR MANAGING A SMART GREENHOUSE

WITH GAMIFICATION ELEMENTS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MARTIN JANEČEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. DAVID BAŽOUT

BRNO 2023

Zadání bakalářské práce



147664

Ústav: Ústav počítačové grafiky a multimédií (UPGM)
Student: **Janeček Martin**
Program: Informační technologie
Specializace: Informační technologie
Název: **Webová aplikace pro ovládání chytrého skleníku s prvky gamifikace**
Kategorie: Webové aplikace
Akademický rok: 2022/23

Zadání:

1. Seznamte se s problematikou řízení podmínek v inteligentním skleníku.
2. Navrhněte intuitivní UI webové aplikace s prvky gamifikace.
3. Návrh testujte s uživateli a proveďte optimalizace.
4. Vyberte vhodnou technologii a proveďte implementaci.
5. Výsledky práce zdokumentujte formou plakátu a krátkého videa.

Literatura:

- Steve Krug: Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability, ISBN: 978-0321965516
- Brian Burke. Gamify: How Gamification Motivates People to Do Extraordinary Things. Routledge (April 24, 2014)

Při obhajobě semestrální části projektu je požadováno:

Body 1, 2, 3 a částečně bod 4.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Bažout David, Ing.**
Vedoucí ústavu: Černocký Jan, prof. Dr. Ing.
Datum zadání: 1.11.2022
Termín pro odevzdání: 10.5.2023
Datum schválení: 3.11.2022

Abstrakt

Cílem této práce je vytvořit intuitivní uživatelské rozhraní pro monitorování a řízení podmínek v chytrém skleníku. Ten je oproti běžnému skleníku doplněn o systém aktuátorů, které za pomoci vzdáleného přístupu z aplikace udržují rostlinám ideální prostředí. Webová aplikace je doplněna o systém gamikace, jehož cílem je motivovat uživatele do využívání všech ovládacích prvků. Gamifikační systém zároveň provází uživatele při prvotním seznamování s chytrým skleníkem. Aplikace je implementována pomocí frameworku Django, prezentační stránka je zajištěna knihovnou React. Vytvořené řešení umožňuje uživateli jednoduše ovládat podmínky v chytrém skleníku a analyzovat historii naměřených dat v grafu či statistikách.

Abstract

The aim of this work is to create an intuitive user interface for monitoring and controlling conditions in a smart greenhouse. This is complemented by a system of devices, compared to a normal greenhouse, which maintain an ideal environment for the plants using remote access from the application. The web application is equipped by a gamification system, which aims to motivate the user to use all the features. At the same time, this system guides the user through the initial introduction to the smart greenhouse. The application is implemented using the Django framework, the presentation side is provided by the React library. The developed solution allows users to easily control the conditions in the smart greenhouse and analyze the history of measured data in a chart or statistics.

Klíčová slova

webová aplikace, chytrý skleník, ovládání, gamifikace, uživatelské rozhraní, Python, Django, JavaScript, React, HTML, CSS, SASS

Keywords

web application, smart greenhouse, control, gamification, user interface, Python, Django, JavaScript, React, HTML, CSS, SASS

Citace

JANEČEK, Martin. *Webová aplikace pro ovládání chytrého skleníku s prvky gamifikace*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. David Bažout

Webová aplikace pro ovládání chytrého skleníku s prvky gamifikace

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Davida Bažouta. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Martin Janeček
5. května 2023

Poděkování

Především děkuji panu Ing. Davidu Bažoutovi za odborné vedení práce a přátelské konzultace. Vřelé díky patří také všem, kteří se podíleli na testování aplikace, a pomohli tak dosáhnout lepších a přesnějších výsledků.

Obsah

1	Úvod	3
2	Problematika chytrého skleníku a gamifikace	4
2.1	Výhody pěstování ve skleníku	4
2.2	Řízení podmínek v chytrém skleníku	5
2.2.1	Teplota a vlhkost vzduchu	5
2.2.2	Vlhkost půdy a zavlažování	6
2.2.3	Koncentrace oxidu uhličitého	6
2.2.4	Světelné záření	7
2.3	Průzkum existujících řešení	8
2.3.1	Sensorie	8
2.3.2	HKGT	9
2.3.3	Avnet	11
2.4	Gamifikace	13
3	Návrh webové aplikace	17
3.1	Cílová skupina uživatelů	17
3.2	Specifikace požadavků	17
3.3	Návrh systému gamifikace	18
3.4	Návrh uživatelského rozhraní	19
3.5	Testování makety	21
3.5.1	Způsob a metriky	22
3.5.2	Výsledky testování	25
3.5.3	Optimalizace po testování makety	27
3.6	Webové frameworky	28
3.6.1	Backend	28
3.6.2	Frontend	29
3.6.3	Shrnutí	33
4	Implementace	34
4.1	Serverová část	34
4.1.1	Databázové modely	35
4.1.2	Koncové body API	37
4.1.3	Testovací data	38
4.1.4	Autentizace	39
4.2	Klientská část	40
4.2.1	Přihlášení uživatele	40
4.2.2	Přehledová obrazovka	40

4.2.3	Řízení podmínek	41
4.2.4	Graf	43
4.2.5	Statistiky	44
4.2.6	Snímky dohledové kamery	45
4.2.7	Systém úspěchů	45
4.2.8	Verze pro mobilní zařízení	46
5	Závěr	48
	Literatura	49
A	Návrh uživatelského rozhraní	52
B	Snímky aplikace	57

Kapitola 1

Úvod

V posledních letech dochází vlivem změn klimatu, a s tím souvisejícími problémy, ke snížení výnosnosti půdy na venkovních plochách. Farmáři i kvůli tomuto problému přechází na pěstování ve sklenících, kde lze přímo pro vybranou plodinu nebo skupinu plodin vytvořit vhodné umělé prostředí. Skleník zároveň chrání úrodu před neodvratnými přírodními vlivy, kterými jsou například přivalové deště či krupobití.

Problematika řízení podmínek ve skleníku není pro laického uživatele příliš triviální. Cílem bakalářské práce je tedy navrhnout a implementovat intuitivní webovou aplikaci pro řízení chytrého skleníku, která běžnému uživateli umožní ovládat komplexní technologii bez hlubších technických znalostí. Koncept chytrého skleníku si klade za cíl předat uživateli produkt, který zaručí kvalitnější a bohatší úrodu, přičemž zároveň sníží čas strávený nastavováním podmínek.

Jako částečný návod k použití je v aplikaci vytvořen systém gamifikace, jenž uživatele motivuje k využívání všech prvků chytrého skleníku. Díky systému odměn lze zjistit, které části aplikace ještě nebyly vyzkoušeny, a uživateli tak především pomůže se v aplikaci zorientovat.

Přehlednost uživatelskému rozhraní dodává ilustrační vizualizace skleníku, která je doplněna o prvky zobrazující aktuální stav podmínek ve skleníku. K regulaci podmínek nedochází z pohledu uživatele tak často, jako k jejich monitorování. Proto byly sekce s nastavením situovány do postranních nabídek, a udrželo se tak čisté a přehledné webové rozhraní.

Lepší pěstitelské úspěchy dokáže zaručit analýza podmínek ve skleníku. V aplikaci je možné procházet historii naměřených dat ze senzorů ve společném grafu a statistické sekci. Při porovnání naměřených dat s kvalitou plodin lze pak snadněji odhalit nevhodně nastavené podmínky.

V následující kapitole 2 je rozvedena problematika řízení podmínek s popisem podstatných faktorů ovlivňujících prostředí ve skleníku a také průzkum existujících řešení. Zároveň je zde podrobněji rozebrána gamifikace. Další kapitola 3 se zabývá procesem návrhu aplikace od specifikace požadavků až po testování makety. Zároveň je zde uvedeno zhodnocení dostupných technologií a návrh systému gamifikace. V následující kapitole 4 je podrobně rozebrána tvorba aplikace. Sekce je prvně zaměřena na serverovou část, jež obsahuje informace o databázových modelech a tvorbě simulovaných dat. Následuje klientská část s popisem prezentační stránky aplikace. V poslední kapitole 5 je shrnuta tvorba webové aplikace, jsou zde hodnoceny výsledky a sepsány náměty na zlepšení.

Kapitola 2

Problematika chytrého skleníku a gamifikace

Podmínky pro pěstování ve skleníku ovlivňuje mnoho přírodních faktorů. Neopominutelnými jsou rychlost a směr větru, koncentrace oxidu uhličitého, sluneční záření, teplota a vlhkost ovzduší. Systém chytrého skleníku monitoruje všechny důležité faktory pomocí senzorů, které zaznamenávají data v reálném čase. Na základě naměřených dat je možné měnit podmínky pomocí aktuátorů, a dosáhnout tak dlouhodobě ideálního prostředí pro pěstování.

2.1 Výhody pěstování ve skleníku

Největší předností skleníků je jejich nezávislost na okolních podmínkách. V umělém prostředí je totiž možné vytvořit ideální podmínky takřka pro jakoukoliv rostlinu. Daným podmínkám navíc vyhovuje více plodin, a pěstitel tak není vázaný jednou konkrétní na celý skleník. Ve sklenicích je díky jejich umělému prostředí velmi jednoduché reagovat na průběžný stav vegetace a měnit si podmínky dle potřeb.

Nespornou výhodou oproti venkovním plochám je delší vegetační doba rostlin a možnost celoročního pěstování. Kontrola podmínek zaručuje ochranu proti náhlým výkyvům teplot, které mohou mít v běžném zemědělství fatální následky. Díky tomu pro pěstování ve sklenicích odpadá pojem sezónní plodina¹. Skleník také slouží jako fyzická bariéra proti nepříznivým povětrnostním podmínkám a silným deštům s krupobitím, které by na otevřených zemědělských plochách mohly výrazně snížit výnosy. [8]

Skleníky jsou rovněž trendem pro ekologické pěstování. Díky izolovanému prostředí je zde výrazně menší riziko nákazy škůdci, a není tak nutné využívat chemické pesticidy. Ty mohou během běžného venkovního pěstování obohacovat půdu o fosfor s dusíkem, a vést tak k nežádoucí eutrofizaci². Dalším plusem pěstování ve skleníku je maximalizace zisku z prodeje. V umělých podmínkách lze dosáhnout výrazně kvalitnějšího ovoce a zeleniny, než při klasickém pěstování, a to i díky prodloužené vegetační době. Je možné díky tomu dosáhnout náskoku před konkurencí a stanovovat si vyšší cenu mimo sezónu. Odpadají zde z velké části také činnosti běžně spojené s venkovním pěstováním, jako je například neustálá starost o růst plevelů či ochranné prvky proti škůdcům a hlodavcům. Z toho vyplývá, že díky skleníku lze ušetřit mnohem více času. [20]

¹plodina, kterou lze sklízet jen během určité části roku

²proces nadměrného zásobování vodního ekosystému živinami, zejména dusíkem a fosforem

Skleník tedy přináší inovativní řešení pro moderní a zdravé pěstování. Šetří peníze i čas a lze díky němu dosáhnout výrazně lepších pěstitelských úspěchů. Koupě skleníku je především investicí do zdravého životního stylu a časové volnosti.

2.2 Řízení podmínek v chytrém skleníku

Řízení podmínek ve skleníku je komplexní technologie, která zahrnuje mnoho faktorů. Následující sekce se zaměřuje na nejdůležitější z nich.



Obrázek 2.1: Základní přírodní faktory ovlivňující podmínky ve skleníku jsou sluneční záření, vítr, relativní vlhkost a vnitřní teplota vzduchu. O řízení těchto podmínek se starají aktuátory, jejichž výčet je zobrazen v pravé části grafiky. Tendencí pro zelenou produkci je tyto elektrické spotřebiče napájet pomocí obnovitelných zdrojů energie. Udržitelnou produkci zajišťuje ve skleníku systém zavlažování a hnojení. [11]

2.2.1 Teplota a vlhkost vzduchu

Teplota vzduchu je jednou z nejdůležitějších proměnných v mikroklimatu skleníku. V chráněných rostlinných systémech probíhá neustálá výměna teploty vzduchu s vnějším prostředím. Únikům tepla přes stěny lze zamezit výběrem masivnějších plastových konstrukcí a střešních krytin, které nepropouští vzduch. Ideální míra vnitřní teploty vzduchu ve skleníku by se pro většinu rostlin měla pohybovat mezi 17 °C a 27 °C. Extrémní rozmezí teplot je 10 °C až 35 °C. Do venkovní teploty 27 °C lze teplotu uvnitř skleníku korigovat přirozeným větráním pomocí otevřených dveří a oken. Při vyšších vnějších teplotách je nutné přijít s umělým chlazením. [7]

Teplotu ve skleníku přímo ovlivňuje také sluneční záření. Množství slunečního svitu přeměněného na teplo závisí na tvaru a orientaci skleníku. Většina přímého slunečního záření je v něm soustředěna na jeho podlahu. Skleník přijímá také difúzní a odražené záření

ze stěn, které taktéž přirozeně zvyšuje teplotu uvnitř skleníku. Správná orientace a tvar skleníku tedy může výrazně snížit náklady na provoz, jelikož je poté možné snížit zatížení instalovaných aktuátorů s vytápěním. [24]

Ke snížení teploty ve sklenících se často využívají mlžící systémy. Ty zároveň zvyšují relativní vlhkost vzduchu, a udržují tak ideální prostředí pro pěstování. Chlazení vnitřního prostředí skleníku je zajištěno odpařováním malých kapiček rozprašované vody. Výhodou zamlžovacích systémů je, že po celém skleníku zajistí rovnoměrné podmínky bez nutnosti větrání. Ochlazení skleníku mohou zajistit také střešní rozprašovače, které kropením stropu skleníku vytváří tenkou vrstvu vodní hladiny, jež svým odpařováním ochlazuje povrch. [18]

Vlhkost vzduchu ovlivňuje různé fyziologické a biologické procesy v rostlinách. Změny vlhkosti vzduchu uvnitř skleníku zajistí především přirozené a umělé větrání. Pro dosažení optimálního růstu plodin je třeba stanovit hodnoty vlhkosti, které musí být ve skleníku udržovány v přijatelných mezích. Hodnoty relativní vlhkosti vzduchu v rozmezí 60 až 90 % jsou pro růst rostlin ideální a nenarušují jejich vývoj. Nižší hodnoty mohou zvláště v teplejších oblastech způsobit vážné problémy kvůli nedostatku vody. Dlouhodobě vyšší vlhkost naopak zvyšuje riziko vzniku houbových chorob. [7]

Pro vypařování nadměrné vlhkosti se dají použít ohřívače vzduchu. Pro skleníky větších rozměrů lze využít vytápění pomocí vody, která se ohřívá v bojleru a poté je rozvedena potrubím po celém prostoru skleníku. Tento způsob ohřevu je ovšem náročný na realizaci i finance. Alternativou pro menší skleníky je vytápění pomocí elektrických ohřívačů. Výhodou je jejich nízká pořizovací cena, jednoduché zapojení i libovolné přemísťování ve skleníku. [6]

Při pěstování více rostlin, které mají odlišné nároky na teplotu, lze skleník doplnit o vyhřívané rohože. Ty se ukládají pod záhony či květináče, a vytváří tak pro určitou plodinu dodatečné teplo. Rohože lze alternativně využívat i pro rychlejší klíčení semen. [13]

2.2.2 Vlhkost půdy a zavlažování

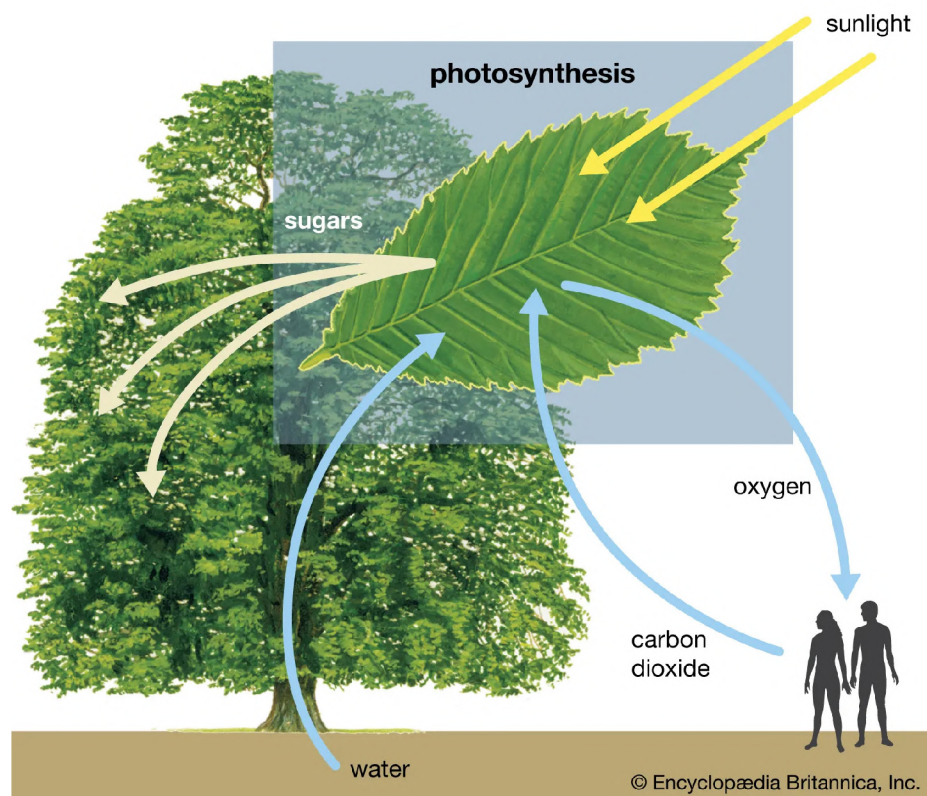
Průběžné měření vlhkosti půdy okolo kořenů je důležité pro optimální míru zavlažování. Většina rostlin potřebuje již při 20 až 25 % vlhkosti půdy doplnit vláhu. Díky půdním senzorům, které udávají poměr vody v substrátu vůči jeho objemu, je možné nastavit prahy zavlažování. Při překročení stanovené hranice se spustí zavlažování určitého vodního okruhu. Pro každý úsek skleníku je vhodné nastavit vlastní potrubní přívod vody, aby nedocházelo k nadměrnému zavlažování dalších okruhů. V praxi se využívají také časovače, které pouští zavlažování pravidelně v zadaném časovém intervalu. Tento způsob zavlažování ale nereaguje na změny vlhkosti půdy. [5]

Zavlažování může doplňovat také stropní rozprašovač. Tento způsob je vhodný zejména pro plochy s velkým počtem sazenic či semen. Rozprašovač produkuje velké množství malých kapek vody a díky tomu dokáže precizněji obsáhnout velké plochy, na kterých pak nehrozí riziko vyplavení nevyklíčených semen. [10]

2.2.3 Koncentrace oxidu uhličitého

Přirozeným způsobem pro udržení hladiny oxidu uhličitého ve skleníku je pravidelná ventilace. Pro obohacení ovzduší oxidem uhličitým lze využít také fosilní paliva, která jej při spalování emitují. Fosilní paliva je možné využít například jako pohonné hmoty pro provoz ohřívačů či jiných spotřebičů. Nejlépe říditelnou možností obohacení CO₂ je přímé

vypouštění plynu z tlakové bomby, jelikož průtok lze nastavit přesně na prahové hodnoty ze senzoru oxidu uhličitého umístěného uvnitř skleníku.³

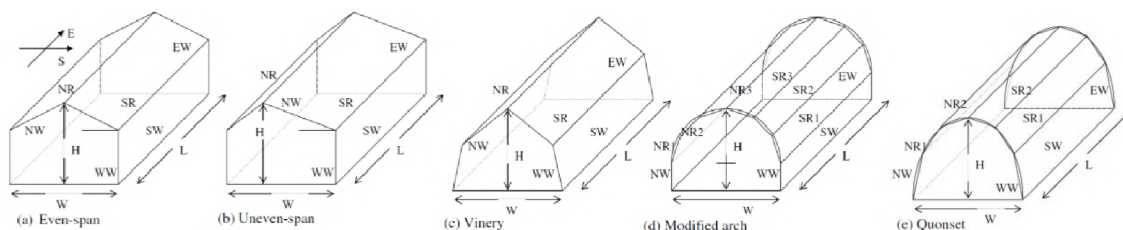


Obrázek 2.2: Oxid uhličitý (CO_2 , angl. *carbon dioxide*) je jedním z klíčových faktorů pro správnou fotosyntetickou funkci rostlin [16]. Ve skleníku je na menším prostoru rozložen velký počet rostlin, proto zde může dojít k nedostatku CO_2 , jelikož jej rostliny pravidelně spotřebovávají k fotosyntéze.

2.2.4 Světelné záření

Světlo lze měřit podle jeho intenzity nebo počtu fotonů dopadajících na povrch. Obecně je v zahradnictví mnohem důležitější počet fotonů, které dopadnou na zemědělský povrch. Počet fotonů zachycených listem určuje míru fotosyntézy a s tím úzce související růst rostlin. Rostliny využívají část spektra zvanou fotosynteticky aktivní záření, což je světlo o vlnové délce 400 až 700 nanometrů. Fotosyntéza dosahuje u rostlin svého vrcholu kolem délky 450 nanometrů, tedy modrého světla, a také při 650 nanometrech, což je vlnová délka červeného světla. Nízká intenzita světla zpomaluje růst rostlin, proto je nutné zamezit stínícím materiálům umístěným nad rostlinami. Při nedostatku světla lze využít umělého osvětlení, které se ovšem vyznačuje vysokou energetickou náročností. Externí osvětlení je možné dodat například i k urychlení klíčení. [21]

³<https://www.youtube.com/watch?v=4ToaP2Ix8AI>



Obrázek 2.3: Množství dopadajícího slunečního svitu na skleník lze zvýšit několika způsoby. Velmi důležitou roli hraje tvar střechy. Ve studii od Sethi [24] byl rozebrán vliv tvaru skleníku na míru slunečního záření. Průzkum ukázal, že skleníky s nepravidelným střechovitým tvarem přijímají maximum slunečního záření během každého měsíce.

2.3 Průzkum existujících řešení

V současné době neexistuje mnoho projektů, které by se zabývaly distribucí chytrých skleníků malých a středních velikostí. Při těchto rozměrech skleníků je to spíše makerská záležitost jednotlivců, kteří si chtějí svůj běžný skleník osadit dalšími aktuátory. Řešení chytrého skleníku menších a středních rozsahů zajišťuje na českém trhu společnost Sensorie⁴.

2.3.1 Sensorie

Podstatnou část průzkumu tvořil reálný chytrý skleník firmy Sensorie, který je doplněn o vlastní webové rozhraní pro řízení. Do aplikace se uživatel přihlašuje pomocí loginu a hesla, při neúspěšném přihlášení se zobrazí varovná textová zpráva. Není zde možnost obnovení hesla či kontaktování podpory.

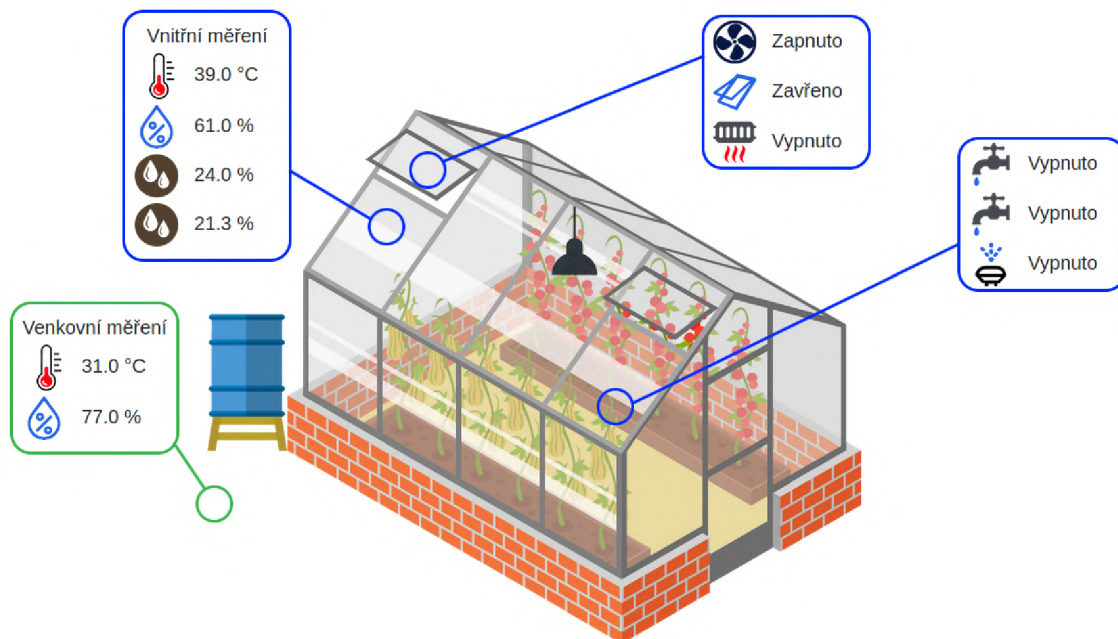
V řídicí sekci aplikace vzdáleného přístupu společnosti Sensorie se pro nastavení podmínek využívají posuvníky se dvěma úchytnými body. Zavlažovací cyklus lze nastavit po určitém čase na předem stanovený zavlažovací interval. Druhou možností je nastavit zavlažování podle vlhkosti, kdy se vlaha udržuje ve stanovených hranicích vlhkosti půdy. Sekce je opatřena také ovládním pro spotřebiče a okna.



Obrázek 2.4: Aktuátory se v řešení firmy Sensorie nastavují pomocí dropdown nabídek. V nich lze vybrat prvně automatický režim, jenž ponechá řízení na automatickém algoritmu, nebo vybraný aktuátor rovnou vypnout či zapnout.

Zákazník si v aplikaci může vykreslovat naměřená data ve třech grafech. První zobrazuje pouze vnitřní a venkovní teplotu vzduchu. Další graf se zaměřuje na zobrazení vlhkostí

⁴<https://sensorie.cz/>



Obrázek 2.5: Aplikace na úvodní obrazovce obsahuje aktuální stavy aktuátorů, zavlažovacích okruhů a střešního okna. Uživatel zde zároveň vidí aktuální hodnoty všech senzorů. Aplikace separuje tuto přehledovou část s řídicí, která je v samostatné kartě.

vzduchu vnitřního i venkovního měření a vykreslení dat z měření vlhkosti půdy. Poslední graf ukazuje stavy aktuátorů v čase. Hodnoty zde nabývají pouze dvou stavů (zapnuto/vypnuto). Uživatel nemá možnost porovnat data napříč těmito třemi grafy.

Aplikace zobrazuje také fotografie z dohledové kamery. Uživatel si může vyselektovat snímky z určitého časového rozmezí a zobrazit je. Režimy snímání lze nastavit na pravidelný, jenž snímá v zadaném časovém intervalu, nebo na snímání při detekci pohybu. Sekce s nastavením kamery je staticky umístěna pod právě vybranou fotografií.

Detektor nemocí rostlin je poslední sekcí webové aplikace od firmy Sensorie. Nahraje se zde fotografie z lokálního úložiště, poté se vybere druh rostliny a fotografie se ořízne do požadovaného čtvercového formátu. Čtvrtým krokem je spuštění detekce. Uživatel musí udělat minimálně čtyři kroky, kterými detekci spustí. Sekce je doplněna textovou nápovědou.

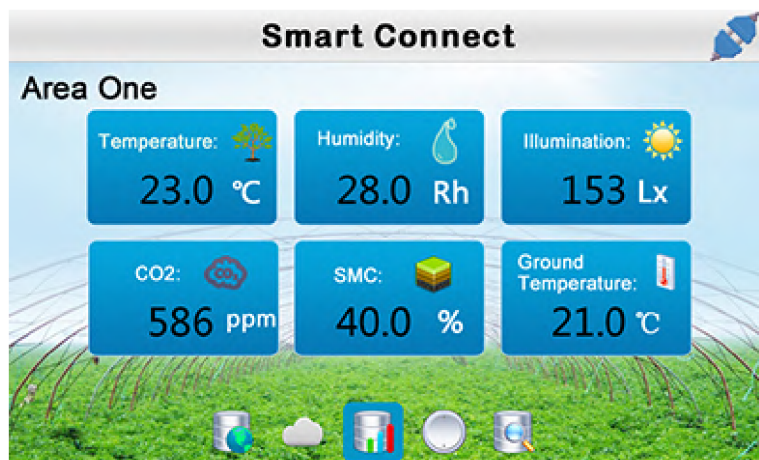
2.3.2 HKGT

Chytrý skleník od společnosti HKGT⁵ si klade za cíl především zdravější a produktivnější zemědělské pěstování. Toho se snaží docílit pomocí systému zařízení, jejichž nashromážděná data pomáhají k nalezení dlouhodobě ideálních podmínek pro pěstování. Veškeré fyzické komponenty chytrého skleníku jsou modifikovatelné a zákazník si je má možnost přizpůsobit podle svých potřeb.

Pro ovládání a monitorování podmínek ve skleníku využívá systém cloudovou platformu Agri-cloud. K platformě lze přistoupit pomocí mobilní aplikace, která zobrazuje všechny měřené hodnoty v reálném čase. Systém řízení je modulární, tudíž se při každé fyzické

⁵<http://www.hkgt.com.cn/content/smartConnect/Eng-smartFarming.html>

komponentě přidá do rozhraní i položka pro její ovládání. Naměřená data systém analyzuje a případně upravuje podmínky ve skleníku pomocí aktuátorů.



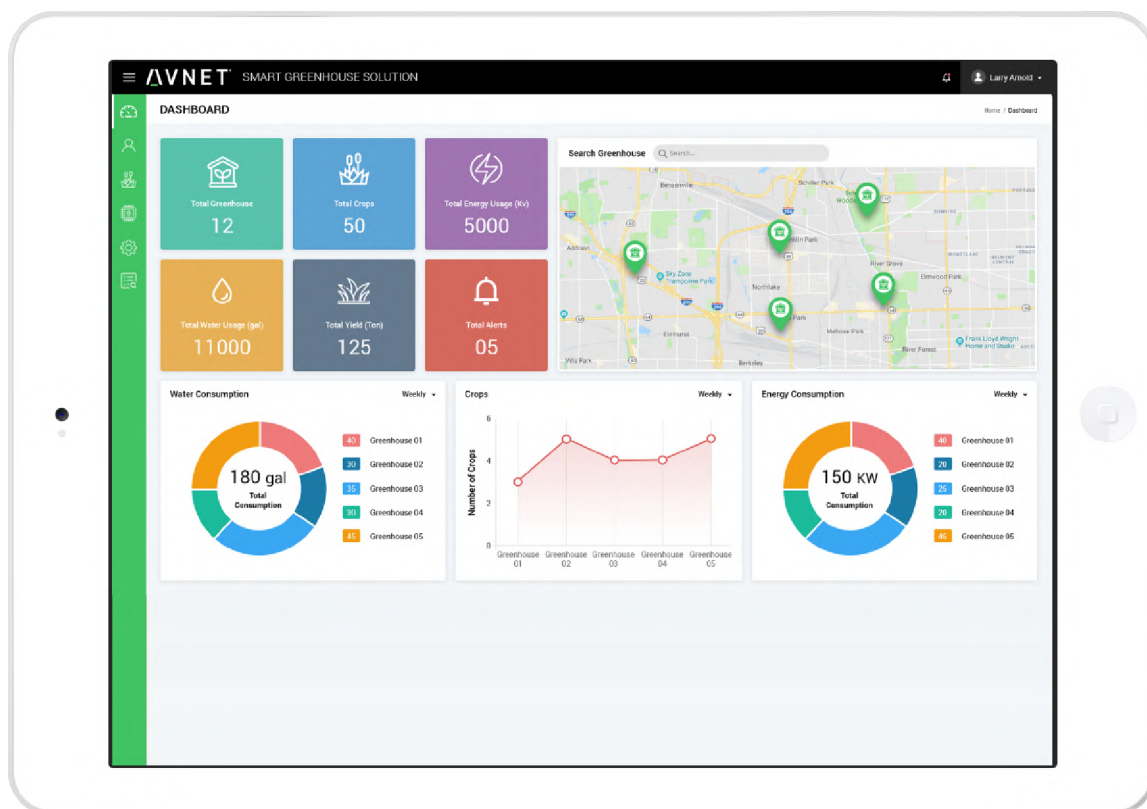
Obrázek 2.6: Běžný systém chytrého skleníku HKGT monitoruje teplotu, vlhkost, intenzitu slunečního svitu, míru oxidu uhličitého a půdní vlastnosti. V hlavní přehledové kartě mobilní aplikace jsou jednotlivé měřené veličiny rozlišeny textem a ikonou.



Obrázek 2.7: Mobilní aplikace má separovanou sekci pro řízení podmínek. Otevírání oken a manipulace se žaluziemi, které zajišťují korekci slunečního svitu vnikajícího do skleníku, se provádí pomocí šipek v pravé části rozhraní. Doplnění vláhy zde zajišťuje postřikovač a kapkové zavlažování. Pro korekci teploty v systému slouží chladicí podložky a ohříváč vzduchu. Dodávají také dodatečný světelný zdroj a ventilátory.

2.3.3 Avnet

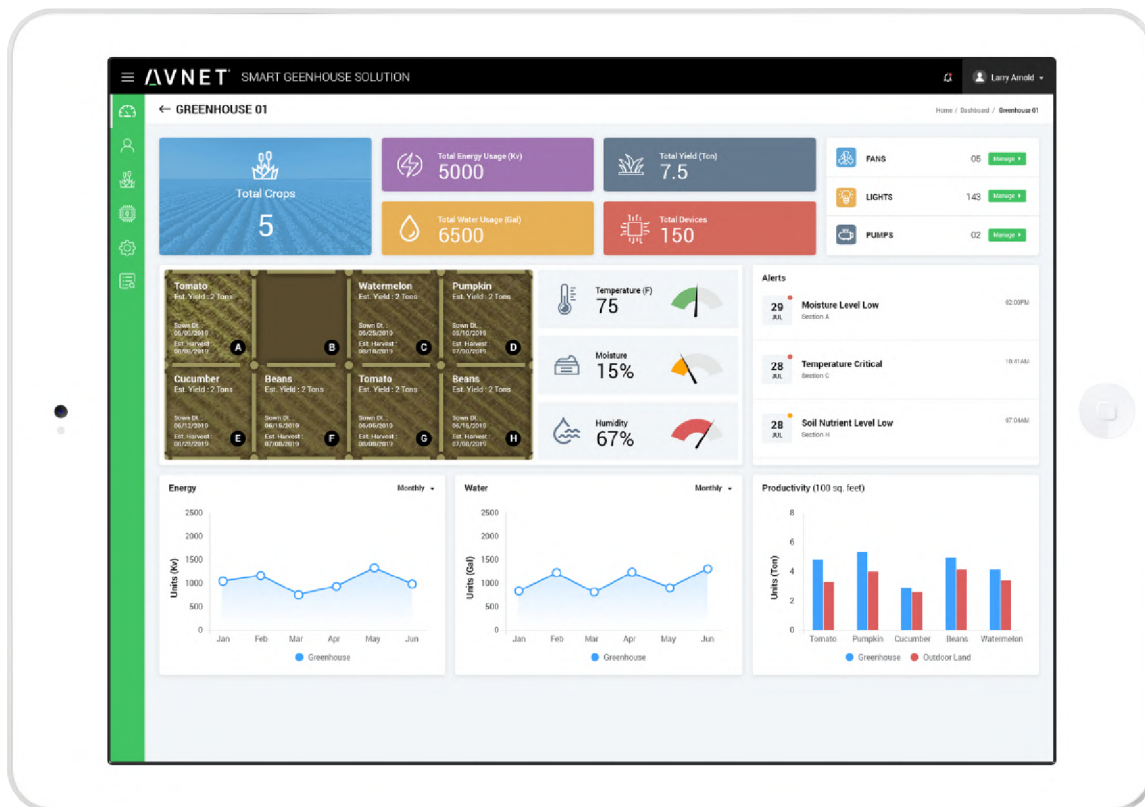
Společnost Avnet⁶ se svým návrhem chytrého skleníku snaží řešit především problém nedostačujících podmínek pro pěstování kvalitních plodin na venkovních zemědělských plochách. Zároveň uvádí, že cílem technického řešení inteligentního skleníku je vyřešit problém přílišné spotřeby elektrické energie.



Obrázek 2.8: Pro monitorování a řízení skleníku od společnosti Avnet je využito grafické rozložení přístrojové desky (angl. *dashboard*). Rozhraní je navrženo pro ovládání více skleníků, figuruje zde i údaj o počtu skleníků a celkové energetické spotřebě. Na úvodní straně lze nalézt údaje o spotřebě vody i elektřiny. Hodnoty jsou uvedené v grafech, kde jsou jednotlivé skleníky rozlišeny barevně. Součástí je také spojitý graf s vývojem počtu plodin. Figurují zde buňky s číselnými hodnotami celkového počtu plodin, spotřebovaných energií, výnosu a upozornění. Každá z buněk je identifikována barvou a ikonou s popisem.

Kromě běžného měření teploty, vlhkosti vzduchu a půdy dodává firma Avnet i monitor fotosyntézy, který měří koncentraci oxidu uhličitého, fotosyntetické aktivní záření, atmosférický tlak i například teplotu listů rostlin. Dále je systém monitoruje také pH půdy, díky čemuž lze korigovat kyselost půdy. Nerovnováha půdních kyselin může snížit dostupnost živin v půdě.

⁶<https://www.iiotconnect.io/smart-greenhouse-solution.html>



Obrázek 2.9: V měřicí sekci pro jednotlivé skleníky jsou uvedeny buňky pro monitorování energetické spotřeby, počtu plodin a zařízení. Je možné se dostat do dalších sekcí a ovládat přímo ventilátory, světla či vodní čerpadla. Své místo v dashboardu mají také elementy pro jednotlivé pěstované plodiny, které v sobě obsahují název, datum sadby, předpokládaný výnos a datum sklizně. Nechybí zde ani grafické elementy pro zobrazení teploty, vlhkosti vzduchu i půdy. Je zde nastaven i systém upozornění, do kterého chodí informace o hraničních hodnotách některého ze senzorů, například při kriticky nízké vlhkosti půdy. Ve spodní části figuruje monitorování energetické spotřeby skleníku.

Aplikace od firmy Avnet cílí také na analýzu plodin. Proto má samostatnou sekci zobrazující stav půdy pro určitou plodinu, což je realizované pomocí kombinovaného spojitého grafu s hodnotami dusíku, fosforu a draslíku v půdě. Zároveň lze dohledat informaci o doporučeném přesazení či očekávané sklizni s predikovaným výnosem.

Součástí aplikace je rovněž detektor nemocí rostlin. Při úspěšné detekci systém zobrazí název nemoci, diagnózu, příčinu a návrhy řešení. Figuruje zde také výpis doporučených produktů s poměrným obsahem účinných látek. Aplikace uživateli vypíše i několik bodů, které slouží jako prevence proti vzniku detekované nemoci. Poslední sekcí aplikace Avnet je řízení a monitorování spotřebičů. V dashboardu je k nalezení spotřeba elektřiny světel s celkovou a průměrnou denní dobou spuštění. Dají se zde nastavit intenzity světel i jejich barevné spektrum. Podobné monitorování je zde i pro zavlažovací okruhy. Průměrná doba a spotřeba se vypočítává pro vodní čerpadla.

2.4 Gamifikace

Gamifikace je využití herních mechanismů mimo klasické herní aplikace s cílem zapojit uživatele více do akce. Pro zapojení lidí využívá herní myšlení, kterým se snaží motivovat k produktivní činnosti či řešení problémů. Často se aplikuje také ve škole, práci a všude tam, kde probíhá hodnocení a snaha o zvýšení pracovní produktivity. Podstatou je gamifikační nástroj ILE (angl. *Interactive Learning Event*), který slouží k vytvoření zábavného prostředí. Tato interaktivní vzdělávací událost může být prezentována výzvami či úlohami, při jejichž plnění předvedou lidé dovednosti v reálném prostředí. Pro gamifikaci se ILE rozděluje do dvou větví, na gamifikaci obsahu a strukturální gamifikaci. [14]

Při **strukturální gamifikaci** (angl. *Structural Gamification*) neprovádí uživatel žádné změny obsahu. Ten není hratelný, ovšem struktura kolem obsahu hratelná je. Primárním cílem strukturální gamifikace je motivace lidí k procesu učení skrze systémy odměn. Typickými elementy v tomto typu gamifikace jsou odznaky, úrovně, mety a žebříčky. Konkrétním příkladem je třeba průběžné získávání odměn za sledování reklamních videí. Naopak **gamifikace obsahu** (angl. *Content Gamification*) se stará o vytvoření zábavných prvků, které jsou hratelné a vtáhnou uživatele přímo do děje. Příkladem mohou být interaktivní kvízy, soutěže či animace herního postupu v mapě. [14]

Využití	Typ ILE
Zvýšení prodeje produktů a služeb	Strukturální gamifikace
Podněcování lidí k využívání aplikace	Strukturální gamifikace
Posílení angažovanosti uživatelů	Strukturální gamifikace
Motivace uživatelů přes zábavný obsah	Obsahová gamifikace
Zlepšení výsledků vzdělávání	Obsahová gamifikace
Podněcování uživatelů k tvůrčí činnosti	Obsahová/strukturální gamifikace

Tabulka 2.1: Využití pro gamifikační interaktivní vzdělávací události.

Gamifikace se dá považovat za jednu z důležitých částí herních mechanismů, ve kterých zaručuje především motivační roli. Hry mají za úkol zejména bavit hráče a pracovat s jejich náladami. Důležitým aspektem jsou kompenzační systémy, jenž mají za úkol odměňovat hráče cenami. Na začátku tvorby systému gamifikace je nutné stanovit si metriky úspěchu. Pro profesní systémy se nabízí metrika peněžní, pro aplikace je to například počet získaných virtuálních odznaků. Druhým bodem v cyklu je vytyčení cílové skupiny. Tvorba gamifikace pro dětskou aplikaci se bude velmi lišit od snahy dosáhnout větší pracovní efektivity zaměstnanců. Důležitým bodem v návrhu je určení správných cílů. Je potřeba si uvědomit, co přesně má být výsledkem gamifikačního procesu. Nejlépe si stanovit jasné kategorie, ve kterých chceme uživatele motivovat, a do těchto kategorií cílit konkrétní úkoly. Typovým příkladem je třeba zvýšení průměrného užívání aplikace uživatelem o půlhodinu denně. S tím je úzce spojen i model zapojení. Uživatele je nutné udržet v systému gamifikace kontinuálně. [4]

Důležitou roli zejména v mobilních aplikacích s prvky gamifikace hraje herní ekonomika. Virtuální herní měnu mohou hráči využívat k nákupu předmětů, které zlepšují herní požitek a mohou díky nim dosáhnout lepších výsledků. Tento způsob gamifikace může vést k monetizaci produktu, kdy se virtuální měna může směnit za reálnou. Pro vývojáře je velmi

složité udělat nákupy za reálnou měnu spravedlivě tak, aby byly dostatečně lákavé, a zároveň nedaly hráči příliš velkou výhodu nad ostatními. [19]

Systém gamifikace nemusí být nikdy konečný. Na základě přístupu uživatelů a testování je vhodné jej modifikovat, kvůli inovaci pro uživatele. Nové prvky dodají systému svěží dech a mohou nastartovat vyšší motivaci. Proto se například mobilní aplikace často aktualizují a jejich interní gamifikační systémy dostávají časem další a další úkoly.

Průzkum existujících gamifikačních systémů

Gamifikační prvky se v dnešní době nachází takřka v každé aplikaci, zejména poté ve hrách. Průzkum byl tedy omezen na mechanismy blízké gamifikačnímu systému pro chytrý skleník.

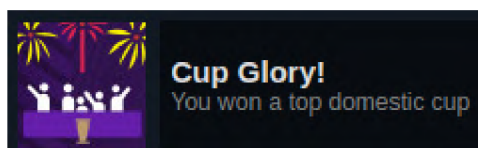
Steam

Platforma Steam⁷ je prostorem pro hraní počítačových her s rozsáhlou podporou komunity. Ve svém obchodu obsahuje více než 30 000 her, několik z nich je zdarma a velká část je placená. Ve službě Steam může uživatel získávat úspěchy pro jednotlivé hry, případně rovnou od Steamu.

Přímo od platformy Steam lze získat odznaky za věrnost, třeba za čtyřleté používání služby. Steam odměňuje uživatele i za počet pořízených her. Systém odznaků v platformě Steam není příliš rozsáhlý. Zajímavým prvkem je však odměna Pilíř komunity zobrazená na snímku 2.11. Ta je tvořena zhruba 30 podúkoly, které hráč dostává za splňování různých akcí. Díky tomuto prvku je možné objevit nepoznaná místa platformy, jelikož zobrazuje i nedokončené úkoly. Po najetí myši vidí uživatel, kde je možné úkoly splnit.

Football Manager 2022

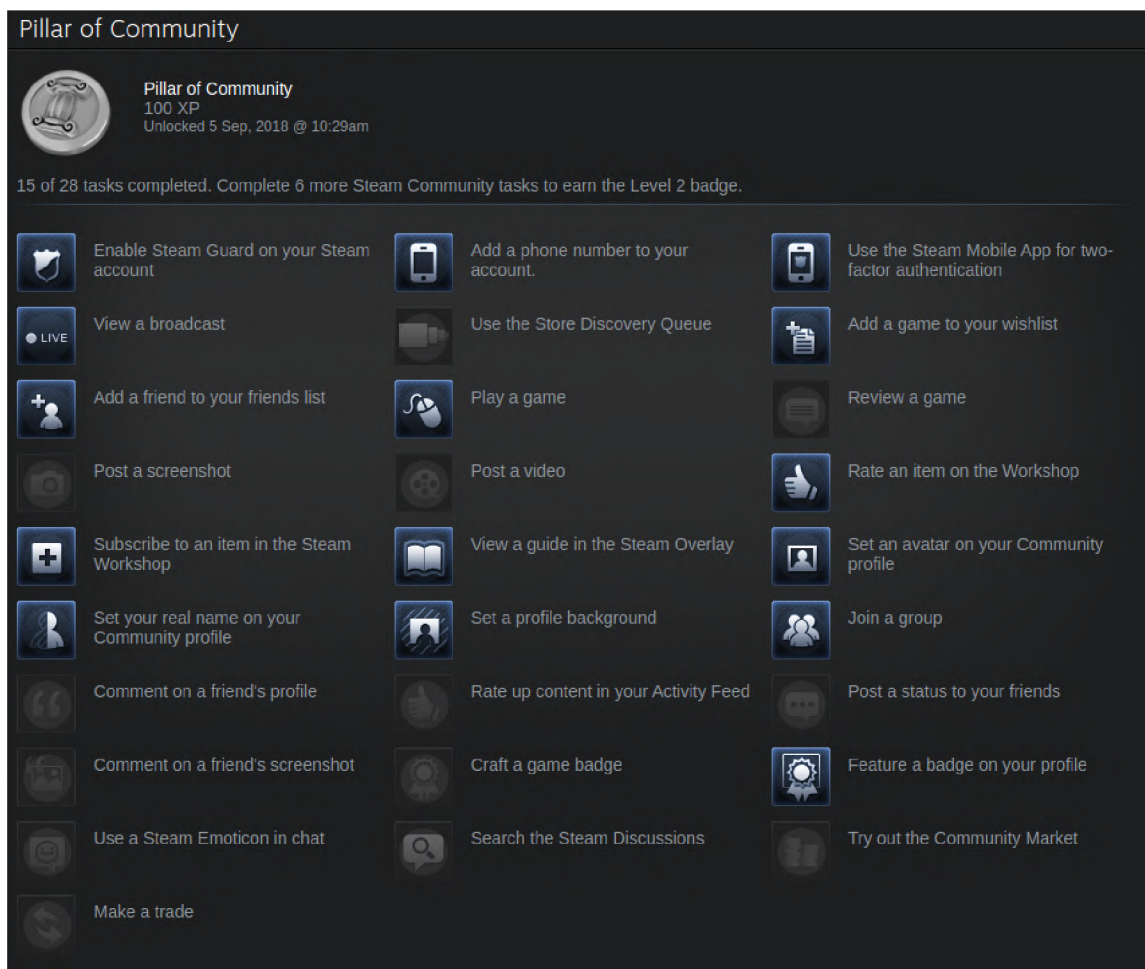
V počítačové hře Football Manager 2022⁸, kterou poskytuje platforma Steam, jsou odměny znázorněny pomocí jednoduchých grafík s titulkem a textem. Uživatel po najetí myši na úspěch vidí kolik procent dalších hráčů daný úspěch získalo, a odhadne tak, jak moc cenný tento úspěch je. Po rozkliknutí úspěchu lze zjistit, kdy byl splněn a jaký byl původní úkol. Součástí je také titulky, který má hráče na první pohled zaujmout, není však nositelem bližších informací. Úkoly ve hře Football Manager 2022 cílí zejména na dosažení úspěchů v samotné aplikaci, nefigurují zde odměny za pravidelné hraní. Jsou zde ovšem úkoly, které realizují postupné navyšování met. Tím je například odměna za počet vyhraných fotbalových duelů. Prvně uživatel obdrží odměnu za pět vyhraných online utkání, následně za 10 a poslední metou je 25 vítězných zápasů.



Obrázek 2.10: Odměna ve hře Football Manager 2022. Odznak je navíc doplněn časovým razítkem splnění. Tento typ strukturální gamifikace motivuje hráče především do získávání co nejvíce odměn pro posouvání se na globálním žebříčku napříč komunitou.

⁷<https://store.steampowered.com/>

⁸<https://www.footballmanager.com/games/football-manager-2022>



Obrázek 2.11: Systém postupných odměn Pilíř komunity v herní platformě Steam.

Tuya Smart

Mobilní aplikace Tuya Smart byla vytvořena pro sdružení chytrých zařízení na jedno místo. Uživatel si tak do jedné aplikace může připojit například chytrou váhu, pračku i osvětlení a ovládat vše pohodlně z jednoho místa. Aplikace funguje pro více značek a kategorií spotřebičů. Jejím cílem je zajistit řídicí centrum pro chytré domácnosti.

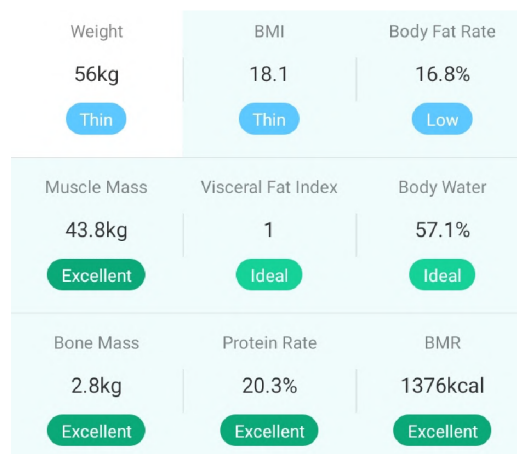
Aplikace má pro každé zařízení jiné funkce. Průzkum je zaměřený pouze na bluetooth chytrou váhu. Ta měří kromě klasické hmotnosti také procentuální zastoupení tuku a svalů v těle, zavodnění těla, hmotnost kostí a mnoho dalšího. Aplikace na úvodní obrazovce 2.12 zobrazí uživateli jeho posun oproti předchozímu vážení, z čehož lze snadno zjistit trend v hubnutí. Po rozkliknutí záložky BMI⁹ (angl. *Body Mass Index*) se uživateli zobrazí měřené statistiky ukázané na obrázku 2.13. Kromě naměřených dat a nadpisu je zde také text s barevným pozadím, jenž zobrazuje stav na stupnici ideální hmotnosti, stavu zavodnění apod. Zelená barva znázorňuje ideální stavy, naopak světle modrá, oranžová a červená značí odchylku od standardu. Tento vizuální aspekt uživateli pomůže ihned najít informaci o svém tělesném zdraví.

⁹index tělesné hmotnosti, určuje zdali má jedinec optimální tělesnou hmotnost

Motivací pro zlepšování svých tělesných stavů je výpočet celkového fyzického skóre. To aplikace automaticky počítá ze všech hodnot. Dále zde figurují přehledné grafy, na nichž je velmi dobře vidět například trend v nabírání svalů. Tento vizuální prvek může uživateli dodat větší motivaci na zlepšení. Do aplikace lze navíc přidat profily dalších uživatelů, kteří tak mohou porovnávat pokrok s ostatními. Do mobilní aplikace je možné zaznamenat cílovou váhu, na kterou se aplikace snaží dostat uživatele právě svými gamifikačními prvky.



Obrázek 2.12: Úsek úvodní strany aplikace Tuya Smart zobrazující poslední naměřené údaje. Strukturální gamifikační prvek je zde realizován do číselné hodnoty značící posun oproti předchozímu měření.



Obrázek 2.13: Statistická sekce jednotlivých hodnot aplikace Tuya Smart. Různé barvy symbolizují porovnání hodnot se standardními tabulkovými hodnotami tělesného zdraví.

Kapitola 3

Návrh webové aplikace

Na začátku návrhu webové aplikace bylo nutné seznámit se s podrobnou specifikací. Po sběru informací z řešerše existujících řešení vznikl návrh uživatelského rozhraní v kolaborativní webové aplikaci Figma¹. Zde byl zároveň vytvořen částečně interaktivní prototyp, který sloužil pro uživatelské testování. To probíhalo s potenciálními uživateli chytrého skleníku a také respondenty, kteří se orientovali v některém z oborů IT, design či agrologie. Na základě sesbíraných dat a nedostatků nalezených při testování byla zapracována optimalizace.

Na konci došlo k průzkumu existujících webových technologií pro vývoj backendu i frontendu aplikace. Z řešerše vyšel pro serverovou část jako nejvhodnější kandidát framework Django, jenž je založený na jazyku Python. Pro frontend byla zvolena prezentační knihovna React, kterou tvoří skriptovací jazyk JavaScript.

3.1 Cílová skupina uživatelů

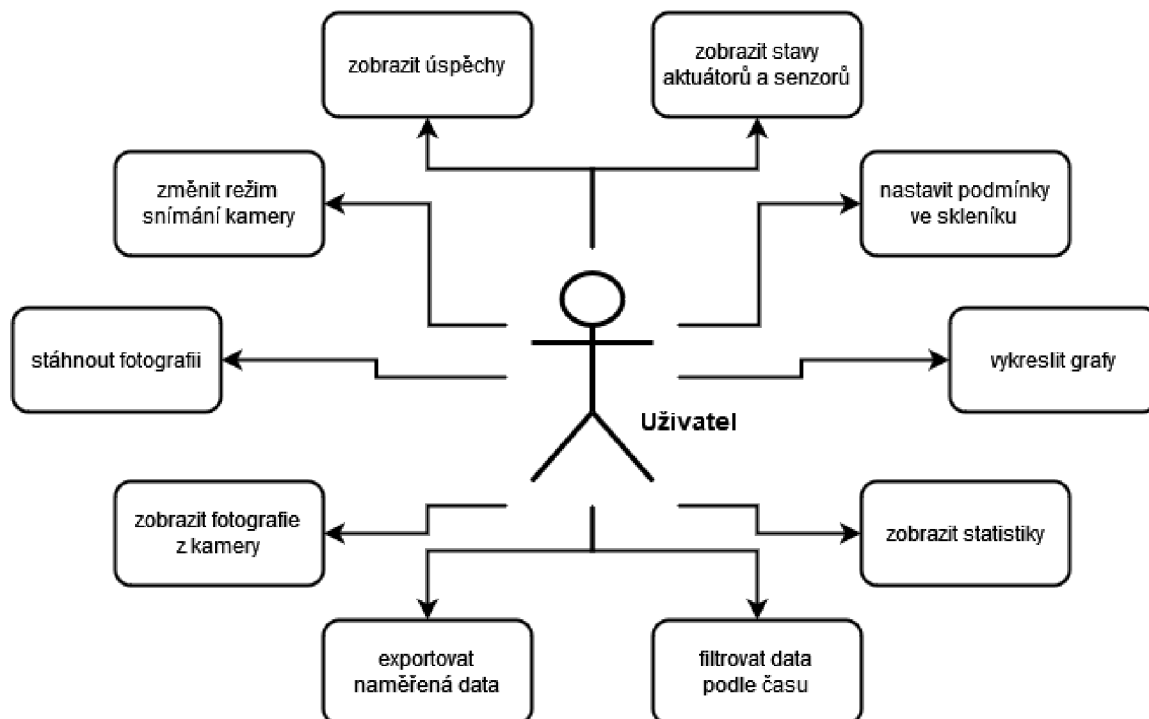
Aplikace je tvořena pro typického uživatele ve věku 30 až 55 let, u kterého se již předpokládá dostatečná finanční rezerva pro koupi chytrého skleníku. Skleníky obecně vyhledávají ekologicky smýšlející lidé, kteří mají zájem o pěstování chemicky nezávadné zeleniny a ovoce. Dále jej využívají lidé, kteří vedou zdravý životní styl, a rádi si tak pěstují zdravé domácí suroviny. Poslední podstatnou skupinou potenciálních uživatelů jsou malé farmy či statky, kteří mohou své vypěstované produkty využívat v lokální restauraci či jako krmivo pro domácí zvířata.

3.2 Specifikace požadavků

Prvotní specifikace požadavků probíhala přímo od vedoucího práce pana Ing. Davida Bažouta. Společnými konzultacemi jsme si ujasnili základní potřeby. Hlavním požadavkem je navržení intuitivního uživatelského rozhraní pro řízení inteligentního skleníku. Na přehledové straně aplikace bylo nutné zobrazit všechny měřené hodnoty a stavy aktuátorů. Tyto hodnoty by uživateli neměly být skryté a měly by být rychle k nalezení.

Ve webové aplikaci funguje systém gamifikace. Uživatel během používání získává úspěchy. Aplikace by měla být responzivní a snadno čitelná i na mobilních zařízeních. Koncept aplikace je potřeba navrhnout multiplatformně, což zaručuje webový systém.

¹<https://www.figma.com/>



Obrázek 3.1: Aplikace umožňuje vykreslení naměřených dat v grafu, které lze exportovat do souboru ve formátu CSV¹⁰ (angl. *Comma-Separated Values*). V aplikaci je potřebné vyhledat informace o době spuštění spotřebičů a ventilů vody. Fotky z dohledové kamery lze volně procházet například podle data pořízení.

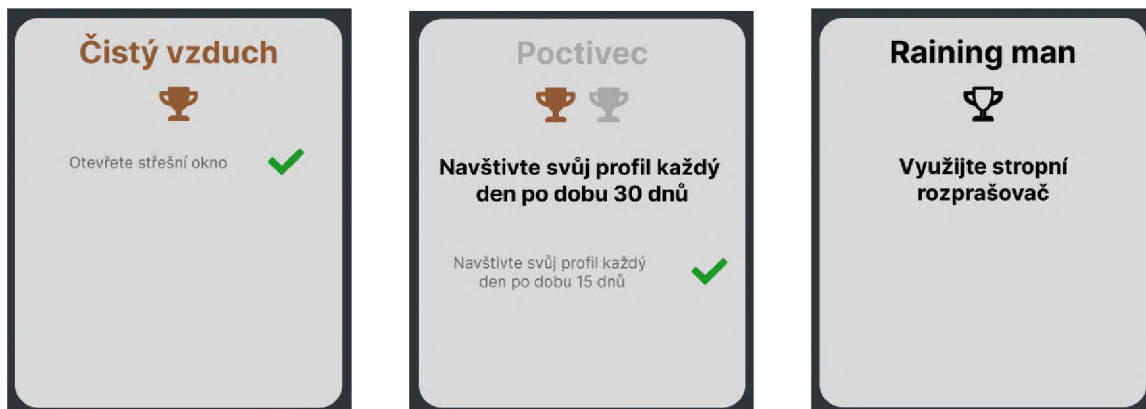
3.3 Návrh systému gamifikace

Hlavním cílem interní gamifikace v aplikaci pro řízení chytrého skleníku je motivovat uživatele do využití všech prvků, které aplikace nabízí. Dílčím cílem je zvýšení pěstitelských úspěchů uživatele. Toho lze docílit pomocí strukturální gamifikace obsahující systém virtuálních odměn. Chytrý skleník si pořizují ekologicky smýšlející lidé, které bude vhodné odměňovat za úspory plastových obalů či úsporu vody. V systému nechybí ani odměňování za pravidelné využívání aplikace či sdílení na sociálních sítích.

Odměny fungují na základě třech stupňů náročnosti. Nejnižší bronzový stupeň odráží základní úroveň ovládnutí skleníku, většinu z těchto úkolů uživatel dosáhne již při prvním seznamování. Ze samotného popisu úkolu lze objevit některé neprozkoumané části skleníku, a systém tak funguje i jako skrytá nápověda. Konkrétním příkladem pro tuto základní úroveň ovládnutí může být například otevření střešního okna ze sekce nastavení. Grafický návrh takového úspěchu je na obrázku 3.2a.

Druhým stupněm v žebříčku úkolů je stříbrný pohár, který lze získat za pokročilejší ovládnutí skleníku. Nejvyšším oceněním je zlatý pohár. Pro neodemčené úkoly slouží pohár s transparentním pozadím a tučně zvýrazněným popiskem úkolu, který je znázorněný na obrázku 3.2c. Tento stav úkolu je na první pohled barevně odlišen od ostatních, a je tedy pro uživatele snadno k nalezení. Je to důležité zejména pro již zmíněné první seznámení se skleníkem, v čem mají právě nedokončené úkoly pomáhat.

¹⁰textový soubor pro ukládání tabulkových dat, kde jsou záznamy oddělené čárkou



(a) Splněný bronzový úkol.

(b) Nesplněná zlatá meta.

(c) Nesplněný úkol.

Obrázek 3.2: Ukázka grafického návrhu úspěchů v různých stavech.

Některé úkoly mohou mít více met. Uživatel může nejdříve získat bronzový pohár a poté se mu ještě stanoví další úkol pro dosažení stříbrného poháru. Každý úkol však vždy začíná na bronzové úrovni. Tento typ je určen zejména pro úkoly, které mají lehce navyšovatelné metriky. Těmi jsou například výnosy v kilogramech. Na obrázku 3.2b je uveden stav, kdy byl splněn bronzový i stříbrný úkol a čeká se na splnění poslední zlaté mety.

3.4 Návrh uživatelského rozhraní

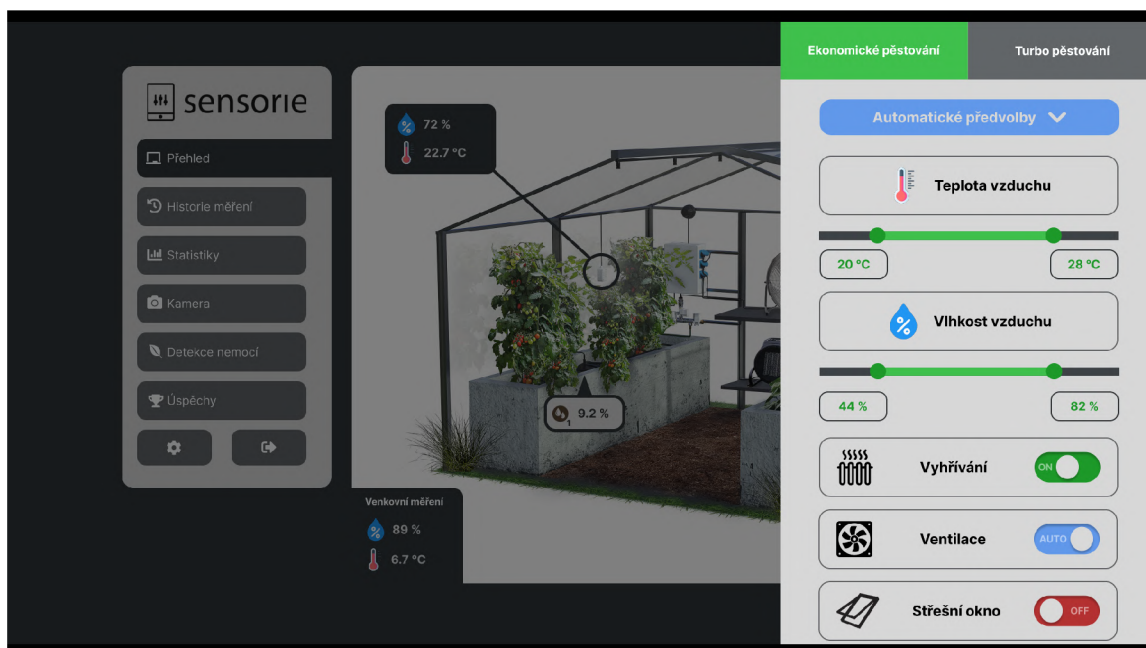
Uživatelské rozhraní by mělo být intuitivní a snadno použitelné. Pokud uživatel musí přemýšlet nad tím, jak se rozhraní používá, tak to znamená, že návrh není dostatečně přátelský. Uživatelé by neměli přemýšlet například nad tím, které z elementů jsou odkazy, díky kterým se lze dostat do cíle. Uživatelé mají omezenou pozornost a budou se snažit najít požadované informace co nejrychleji. Proto by měla být aplikace navržena tak, aby důležité informace byly snadno viditelné a rychle dostupné. [15]

Návrh webové aplikace pro ovládání chytrého skleníku byl vytvořena pomocí aplikace Figma, ve které lze začlenit také omezené interaktivní prvky vhodné pro uživatelské testování. Drátěný model² nebyl pro návrh dostačující, jelikož by z něj oproti částečně funkčnímu prototypu nebylo možné získat tolik relevantních dat.

Přihlášení a přehledová obrazovka

Při vstupu do webové aplikace uvítá uživatele přihlašovací obrazovka A.1 s loginem a heslem. Po úspěšném přihlášení se uživatel dostane na základní přehledovou obrazovku, kde se na vizualizaci obecného skleníku zobrazují všechny měřené údaje a stavy aktuátorů. Probíhá zde také veškeré řízení podmínek, které je skryté v postranní nabídce.

²základní náčrt uživatelského rozhraní, jenž neobsahuje barvy, obsah ani interaktivitu



Obrázek 3.3: V levé části návrhu figuruje základní menu, aktivní sekce je znázorněna tmavší barvou. Po kliknutí na tmavé elementy s měřenými daty na vizualizaci skleníku se otevře nabídka nastavení. Cílem bylo skrýt před uživateli veškeré nastavení mimo přehledové sekce. Nastavení rozmezí dat je navrženo pomocí posuvníku se dvěma úchytnými body. Údaje je možné nastavit také pomocí textového pole. Řízení aktuátorů probíhá třístavovými tlačítky. V návrhu je i automatické přednastavení podmínek podle druhu plodiny.

Historie měření

Sekce Historie měření obsahuje jeden graf, který je na obrázku A.3. Je zde možné vykreslit jakékoliv měřené veličiny. Pro přehlednost jsou zde veličiny seskupeny do čtyř kategorií, kterými jsou teploty a vlhkosti, půda, zařízení a zavlažování. Uživatel má možnost jedním kliknutím zobrazit všechny měřené veličiny z určité kategorie, nebo je naopak všechny skrýt.



Obrázek 3.4: Data pro zobrazení se vybírají pomocí dvou hraničních dat v kalendáři. Hodnoty ve vybraném rozmezí je možné exportovat prostřednictvím CSV souboru. V návrhu je také možnost importu dat.

Statistiky

Sekce Statistiky zobrazuje uživateli dobu spuštění spotřebičů a zavlažovacích okruhů. Snímek celé obrazovky této sekce je obrázku A.4. Zajímavými údaji může být průměrná spotřeba elektřiny a vody s následným výpočtem finančních nákladů na provoz chytrého skleníku. Data se vybírají v určitém časovém rozmezí pomocí kalendáře. Součástí je i plánovaný zápisník pěstitele, jeho obsah je předmětem uživatelského testování makety.



Obrázek 3.5: Návrh zobrazení statistických dat. V sekci se kromě doby spuštění aktuátorů zobrazují také průměrné spotřeby elektriny a vody.

Kamera

Část s kamerou, uvedená na obrázku A.5, tvoří fotografie ze skleníku. Při vstupu do sekce Kamera se zobrazí vždy poslední pořízená fotografie. Mezi snímky lze procházet pomocí šipek, pro přechod ke starším snímku slouží kalendář. Fotografie v náhledu je možné stáhnout. Nastavení snímání kamery je opět přesunuté do postranní nabídky, která se rozbíjí po kliknutí na ikonu nastavení. Zde se pomocí dvoustavového přepínače volí způsob snímání. Figuruje zde posuvníky a pole, kam lze napsat přímo z klávesnice přesnou číselnou hodnotu. Postranní nabídka také nabízí hromadné uložení snímků ze zadaného časového rozmezí.

Detekce nemocí

Vstup do detektoru nemocí je navržen velmi jednoduše. Pro nahrání fotografie stačí kliknout na tlačítko nebo ji přetáhnout pomocí myši na určené místo. Po nahrání se zvolí druh rostliny a spustí se samotná detekce. Výsledkem je grafický oddíl obsahující název nemoci, popis, řešení a prevenci. Je zde také proklik přímo do plánovaného atlasu nemocí. Ukázkou detektoru před vložením snímku ukazuje obrázek A.6, snímek A.7 je po dokončení detekce.

Úspěchy

Poslední sekci, jejíž snímek je na obrázku A.8, z hlavní nabídky jsou Úspěchy. Zde se zobrazují všechny úspěchy napříč aplikací, tedy i ty nedosažené. Každý úspěch je samostatná komponenta skládající se z názvu, stupně úspěchu, popisu úkolu a označení splněných částí. Stupně úspěchu zde realizuje systém třech pohárů, jimiž jsou bronzový, stříbrný a zlatý. Zlatý pohár zde označuje maximální dosažitelnou úroveň. Nesplněné úkoly mají černou barvu a v hlavičce elementu obsahují pohár s transparentní výplní. Splněný úkol uživatel pozná díky zelené fajfce u popisu úkolu. Neaktivní úkol zároveň zašedne a zmenší velikost textu, aby v uživateli nebudil větší pozornost.

3.5 Testování makety

Návrh uživatelského rozhraní je nutné vylepšovat na základě zpětné vazby od uživatelů. Testování by se mělo zaměřovat na klíčové prvky rozhraní a mělo být prováděno již v raných fázích návrhu, aby se odbouraly základní nesrovnalosti. Při testování s reálnými uživateli je nutné sledovat jejich postup a interakci s prostředím. Testování má být

záživné a motivující, mělo by být prezentováno s cílem vylepšit uživatelské rozhraní podle svého, nikoliv jako zátěž či stresová aktivita. [15]

Testování makety vytvořené v aplikaci Figma se účastnili lidé, kteří se splňovali alespoň jedno z těchto kritérií:

- studijní/profesionální zaměření v IT, designu či agrologii
- věkové rozmezí 30 až 55 let

Lidé z IT oborů či designu přispívali technickými poznatky ke způsobu implementace. Testování z oblasti agrologie především upřesňovali potřeby reálného pěstitele.

3.5.1 Způsob a metriky

Na začátku testování makety byly zaznamenány údaje o respondentovi. Do formuláře bylo zapsáno jméno, věk, profesionální nebo studijní zaměření, zájmy, úroveň práce s počítačem a informační gramotnosti. Posledním měřítkem byla míra zkušeností s testováním prototypů.

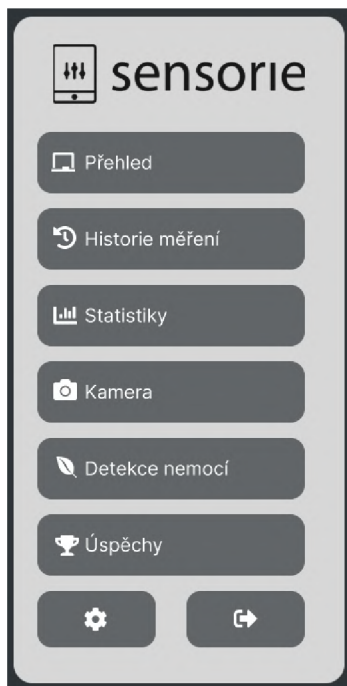
Způsobem testování byl řízený rozhovor nad částečně interaktivním prototypem, u kterého byla nahrávána obrazovka i se zvukovým záznamem. V úvodu byly vysvětleny důvody testování návrhů a jaké výhody to přinese pro implementaci. Podle potřeb a znalostí uživatele bylo objasněno jakým způsobem lze maketou procházet a kde by mohl očekávat interaktivní prvky. Poté byl respondent seznámen se všemi podstatnými elementy chytrého skleníku. Testování makety lze logicky členit do třech částí. První fází bylo přiřazování kartiček k sekcím v menu, poté testování přešlo v průchod aplikací s plněním úkolů podle jednotného scénáře. V poslední fázi se testovaly některé komponenty návrhu, srozumitelnost ikon a ovládacích prvků.

Přiřazování kartiček

Na úvod byl uživateli zobrazen pohled se základním menu a vizualizací obecného skleníku. Zde došlo k první fázi testování, kterým bylo přiřazování pomyslných kartiček k jednotlivým sekcím v menu. Při této metrice se přiloží testovanému několik kartiček s textovými informacemi, které má za úkol přiřadit právě do jedné z předem určených kategorií. Kartičky byly v testování pomyslné a uživatel přijímal informace pouze slovně. Cílem metodiky je porozumět běžným uživatelům při kategorizování informací, tedy kde přesně by očekávali jednotlivé sekce webové aplikace.

Metodiku přiřazování kartiček tvořilo pro každého respondenta 25 úkolů. Úkoly byly nastaveny tak, aby pokryly nejčastější akce uživatele při řízení a monitorování chytrého skleníku. V odřázkách níže se nachází ukázka pěti úkolů.

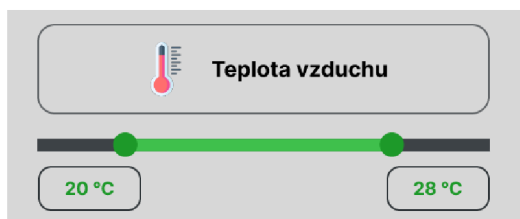
- V jaké sekci naleznete nastavení teploty?
- V jaké sekci si zobrazíte spotřebu elektrické energie?
- V jaké sekci si v grafu zobrazíte venkovní vlhkost vzduchu?
- V jaké sekci vypnete ventilátor?
- V jaké sekci stáhnete poslední pořízenou fotografii z dohledové kamery?



Obrázek 3.6: Návrh základního menu aplikace, který posloužil testování metodikou přiřazování kartiček. Informacemi zde byly jednotlivé části aplikace, kategoriemi poté sekce ze základního menu.

Průchod prototypem s plněním úkolů

Druhou část testování tvořilo celkově 24 hlavních úkolů podle předem připraveného jednotného scénáře v částečně interaktivním prototypu aplikace Figma. Na většinu z nich byla aplikována časová metrika, ostatní vyžadovaly pouze slovní reakci. Uživatel u všech sekcí komentoval první dojem a přehlednost. Testovaný byl v průběhu systematicky vybízen k jakýmkoliv podnětům na zlepšení návrhu.



(a) Řízení teploty vzduchu pomocí táhla se dvěma úchytnými body a textového pole.



(b) Řízení aktuátorů pomocí třístavových přepínačů.

Obrázek 3.7: Návrh řízení podmínek v postranním rozbalovacím menu.

Pro úkoly s vyhledáním informací se řešil především čas potřebný na splnění. Důležitými podněty byly také slovní reakce během orientace v prostředí. V sekci nastavení bylo nutné zjistit, zdali je navržený způsob selekce dat pomocí táhla a textového pole dostatečně intu-

itivní. Důležitou roli zde hrálo také testování neobvykle navrženého třístavového přepínače pro změnu stavů aktuátorů.

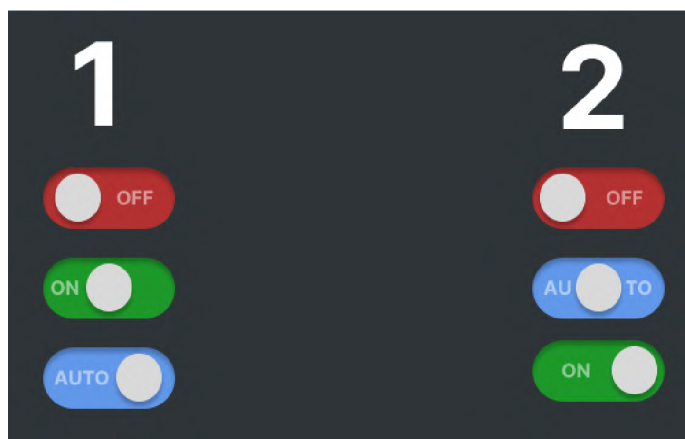
Při testování v sekci Historie měření respondent hodnotil logické členění dat v legendě a jejich barevné rozlišení. Poté v grafu zobrazoval různé hodnoty. Ve statistické sekci probíhala s uživateli zejména diskuze o statistických datech ze skleníku, která by byla pro řízení potřebná. Stejně tak byl testovaný dotázán, které informace by si mohl zapisovat do virtuálního zápisníku pěstitele.

V sekci s kamerou uživatel zobrazoval fotografii z určitého data, sledovaným faktorem byla zejména rychlost a také způsob, kterým chtěl dosáhnout cíle. Diskutováno bylo nastavení režimu snímání, kde uživatel měnil režim na pravidelné snímání. V detektoru nemocí rostlin uživatel nahrál fotografii a měl za úkol dokončit proces simulované detekce zjištěním nemoci. Nakonec hodnotil, které faktické údaje by uvítal ve výpisu po dokončení detekce.

Poslední sekci druhé fáze testování makety byly úspěchy. Zde byl testovanému krátce představen systém gamifikace a jeho role v rámci webové aplikace. Následně měl zhodnotit systém pohárů reprezentující tři kategorie. Poté měl mezi několika úspěchy na obrazovce vybrat jen nesplněné. Následně měl vyhledat i úspěchy, které ještě nedosáhly finální fáze. Cílem tohoto typu testování bylo zjistit míru porozumění symbolům a barevným odlišením v rámci jednotlivých úspěchů. Následně se uživatel z aplikace odhlásil.

Test komponent

V poslední fázi testování makety měl respondent za úkol projít některé komponenty aplikace a vyzkoušet na nich separovaně jejich uživatelskou přívětivost. V první části se zaměřil na třístavové přepínače. Navrženy byly dvě logické verze třístavových tlačítek a respondent měl rozhodnout, které z nich je podle něj přívětivější. Tlačítka v maketě byla plně interaktivní, jejich stavy tak mohl respondent volně měnit.



Obrázek 3.8: Rozdíly mezi tlačítkami byly v poloze stavů ON a AUTO. Stav OFF má svou polohu vždy nalevo. Jedna verze tlačítka byla v poloze uprostřed ve stavu ON, v poloze napravo poté ve stavu AUTO. Druhý typ tlačítka měl tyto stavy prohozené.

Následoval test dvoustavových tlačítek, jehož cílem bylo zjistit, jak uživatelé detekují aktivní a neaktivní stavy přepínačů. Poté byly testovanému zobrazeny některé ikony z návrhu. Úkolem pro respondenta bylo vysvětlit význam ikony a jakou akci by očekával při jejím stisknutí pomocí levého tlačítka myši. Díky tomu lze získat důležitá data o vnímání ikon z pohledu běžných uživatelů.

3.5.2 Výsledky testování

Celkově podrobným testováním makety prošlo sedm respondentů, kteří pokryli všechny typické uživatele. Testování odstartovali dva respondenti z věkové kategorie 30 až 55 let, kteří byli na nižší úrovni informační gramotnosti a neměli s maketami žádnou zkušenost. Jeden z nich spadl do technického profesního zaměření, tudíž se v částech chytrého skleníku začal orientovat rychleji. Druhý respondent měl zprvu problémy i se základním ovládním aplikace v nabídce. Účastnil se také student agronomické fakulty se zájmy o biologii a chemii, který měl již zkušenost s testováním makety. Ostatní lidé byli studenti technických oborů se zájmy o IT a techniku. Jejich úroveň informační gramotnosti a práce s počítačem byla na vysoké úrovni, testování tedy neomezovaly nezkušenosti s běžnými ovládacími prvky webových aplikací. Dva z nich v minulosti sami tvořili maketu přímo v aplikaci Figma.

1. část - Přiřazování kartiček

Přiřazování kartiček odhalilo u všech testovaných problém s orientací v hlavní nabídce. Při rozdělování imaginárních kartiček k sekcím v menu dělalo uživatelům velký problém rozlišit sekce Historie měření a Statistiky. Informace mezi těmito sekcemi křížili a slovně komentovali nevhodnost pojmenování těchto sekcí. Úspěšnost přiřazení k těmto dvou sekcím byla oproti celkové statistice výrazně podprůměrná. Návrhem od testovaných bylo například sjednocení sekcí do jedné či přejmenování Historie měření na Graf. U statistické sekce je navíc nevhodně zvolená ikona grafu, jelikož graf s měřenými hodnotami obsahuje jen sekce Historie měření. I proto většina respondentů očekávala graf v sekci Statistiky.

Sekce	Úspěšnost přiřazení
přehled	91,07 %
historie měření	28,57 %
statistiky	54,29 %
detekce nemocí	94,24 %
kamera	90,48 %
úspěchy	100 %
celková úspěšnost	77,71 %

Tabulka 3.1: Ze statistiky úspěšného přiřazení informace k sekci v menu vychází, že většinu informací z přehledové sekce přiřadili respondenti správně. Přehledová sekce aplikace je obecně nejnavštěvovanější, když obsahuje všechna aktuálně monitorovaná data. Součástí sekce je také postranní navigace s nastavením, jejíž snadné a intuitivní ovládní je klíčové.

2. část - Průchod prototypem s plněním úkolů

Při úkolu s nastavením dolní hranice teploty vzduchu byl po splnění respondent dotázán, jestli pro výběr dat preferuje táhlo (angl. *slider*) nebo textové pole. Testování z věkové hranice 30 až 55 let preferovali zadání hodnoty přímo číslem do pole. Zbýlých pět otázek bylo hodnota teploty vzduchu nastavilo pomocí slideru. Všichni ovšem označili za vhodné ponechat obě varianty, tak jak bylo navrženo v prototypu. Jeden z respondentů se zaměřil na skok, kterým se slider posouvá, a navrhoval jej minimálně o 0,5 stupně, jinak by pro něj mohlo být selektování frustrující.

V sekci nastavení se mezi respondenty osvědčilo třístavové tlačítko, které hodnotili jako intuitivní. Výhodou byla velmi rychlá selekce stavů, kterou lze vypořádat i z naměřených časových údajů pro úkoly v tabulce 3.2. Řadu respondentů mátl stav vodního čerpadla, jehož design byl navržen podobně jako interaktivní třístavové tlačítko, přičemž nebylo přepínatelné.

V sekci Statistik byli respondenti otázaní na údaje, které by v této sekci rádi zobrazovali. V aplikaci by ocenili možnost i možnost nastavení vlastní ceny za elektřinu a vodu. Nejčastějšími odpověďmi byly:

- Cena a celková spotřeba elektřiny
- Cena a celková spotřeba vody
- Spotřeba energie za osvětlení
- Spotřeba elektřiny pro jednotlivé spotřebiče
- Doba zapnutí jednotlivých spotřebičů

Řada respondentů shledávala grafickou reprezentaci elementů pro úspěchy jako zmatečnou. Odráželo se to zejména při druhé fázi testování, kdy měli vyhledávat úspěchy v různých stavech. Na první pohled pro ně často nebylo zřejmé, jestli je daný úspěch dokončený či má ještě další metu. Systém zelené fajfky pro dokončené dílčí mety většina testovaných nechápala. Nápadem bylo například využít pouze jednu barvu pohárů a jejich úroveň stanovovat počtem získaných pohárů. Respondenti z věkové hranice 30 až 55 let měli s pochopením systému odměn největší problém, jelikož se s gamifikačními prvky nepotkávají. Jeden z nich navrhoval, aby se pro jednoduchost zrušily navazující úkoly. Testování z věkové hranice do 26 let považovali systém o třech stupních a dílčích úkolech za vhodný, problém byl dle nich jen ve způsobu grafické reprezentace, která nebyla dostatečně uživatelsky přívětivá.

Mezi kladně hodnocené části prototypu patří zejména umístění všech sekcí s nastavením do bočního panelu. Plusem při seznamování s chytrým skleníkem se ukázaly také popisky jednotlivých částí skleníku při najetí myši. Dva testování by ovšem ocenili, kdyby se akce po najetí myši daly postupem času vypnout. Obecně dobrým dojmem působil graf v sekci Historie měření. Respondenti ocenili způsob zadávání hodnot pomocí kliknutí na popis v legendě a jeho následné zbarvení či zašednutí dle aktivity dat v grafu. Kladného zhodnocení se dočkalo i třístavové tlačítko pro řízení aktuátorů. Uživatelé na první pohled poznají stav tlačítka díky dvěma aspektům, barvě a popisu, a nevadilo jim tak neobvyklé přepínání mezi třemi stavy namísto běžných dvou.

Úkol	Výchozí sekce	Doba plnění
Zjistěte vnitřní teplotu vzduchu	přehled	4,29 s
Zjistěte stav ventilátoru	přehled	11,43 s
Zjistěte vlhkost půdy 1. okruhu	přehled	4,29 s
Zjistěte venkovní vlhkost vzduchu	přehled	5,14 s
Zobrazte na modelu rozprašovač	přehled	10,29 s
Zjistěte zdali je zapnutý rozprašovač	přehled	2,43 s
Nastavte dolní hranici teploty vzduchu na 17 °C	přehled	20,29 s
Dostaňte se do nabídky spotřebičů a střešního okna	přehled	12,29 s
Otevřete střešní okno a vypněte ventilaci	nastavení	5,00 s
Nastavte zavlažovací okruh 1 na AUTO	přehled	9,57 s
Změňte zavlažování 2. okruhu podle vlhkosti	nastavení	3,86 s
Zjistěte jestli je zapnuté vodní čerpadlo	nastavení	4,14 s
Zobrazte v grafu všechny hodnoty se sekce zařízení	historie měření	9,57 s
Zobrazte v grafu vnitřní teplotu vzduchu	historie měření	7,43 s
Zobrazte fotografii z 5.12.2022 17:00	kamera	3,71 s
Stáhněte aktuální fotografii	kamera	3,00 s
Změňte režim na pravidelné snímání	kamera	8,14 s
Ukažte na úspěchy, které ještě nejsou splněné	úspěchy	6,29 s
Odhlaste se z aplikace	úspěchy	5,71 s

Tabulka 3.2: Průměrný čas úspěšného splnění úkolů s časovými metrikami pro druhou část testování makety.

3. část - Test komponent

V úvodu poslední části testování prototypu volili potenciální uživatelé vhodnější typ třístavového přepínače. Celkově šest ze sedmi respondentů zvolilo jako intuitivnější verzi číslo 1 z variant uvedených na obrázku 3.8. Důvodem pro první verzi byla především potřeba přejít z vypnutého stavu přímo do zapnutého, jak je ve fyzických spotřebičích běžné. Automatický režim vnímali jako stav navíc, tedy by dle zpětné vazby měl figurovat až jako poslední možnost.

Testovaným dělala problém ikona pro vlhkost půdy, která na první pohled tuto veličinu nesymbolizovala. Hnědá barva byla sice dobře zvolená, lepší představu o ikoně by dodal ještě například symbol procenta pro relativní vlhkost. Horší na orientaci byla zároveň ikona vodního čerpadla. To především z důvodu, že jej lidé v běžném životě příliš nepotkávají. Negativní reakce přišla i na ikonu ventilátoru, která se tematicky nehodila k ostatním v přehledu. Dva respondenti tuto nuanci označili za frustrující.

3.5.3 Optimalizace po testování makety

Zpětná vazba z testování byla zapracována již v průběhu implementace. Další optimalizovaný návrh nebylo nutné tvořit, jelikož se navržený koncept aplikace během testování prototypu osvědčil, a změn tak nebylo mnoho. Nejdůležitější změnou bylo přejmenování sekce Historie měření na Graf. Jak již vyplynulo z procentuální úspěšnosti přiřazování kar-

tiček, respondenti tuto část hojně zaměňovali se sekci Statistiky. Zároveň došlo ke změně nevhodných ikon pro aktuátory.

Elementy s daty na přehledové obrazovce neměly v návrhu jednotnou strukturu, část s vnitřní teplotou vzduchu měla u sebe ukazatel na senzor teploty a vlhkosti vzduchu na vizualizaci. Po optimalizaci se všechny tyto elementy umístily do rohů. Pro vnitřní a venkovní měření byly doplněny popisky, které některým respondentům chyběly.

Postranní menu z nastavením se ukázalo jako velmi intuitivní a po zpětné vazbě bylo akorát doplněno tlačítko na zavření nabídky. Sekce Statistiky byla doplněna o všechna měřená data, která uživatele zajímala ve druhé části testování. Zejména se tak aplikace doplnila o statistiky spotřeby elektřiny a vody.

Sekce s kamerou byla zejména pro respondenty z věkové kategorie 30 až 55 let méně přehledná, a to kvůli nepřehledným ovládacím prvkům umístěným přímo na fotografii. Po optimalizaci figurovaly šipky pro procházení fotografií a tlačítko na stažení aktuálního snímku mimo vybraný snímek.

Po zpětné vazbě ze třetí části testování se úprav dočkaly i některé komponenty a nejasné ikony. Sekce vodního čerpadla v postranní nabídce nastavení měla pouze informační charakter, proto byl celý element s vodním čerpadlem odstraněn. Změněna byla i ikona ventilátoru, která byla vyhodnocena jako tematicky nevhodná ke zbytku ikon.

Kompletní změnu zažil systém gamifikace. Po nepříliš intuitivním systému, který naplno pochopili pouze lidé se zkušenostmi z již existujících herních systémů odměn, byl vytvořen mnohem jednodušší systém. Původní tři barvy pro poháry byly zrušeny, nahradila je pouze zlatá barva. Náročnost jednotlivých úspěchů byla stanovena podle počtu zlatých pohárů.

3.6 Webové frameworky

Frameworky poskytují programátorovi sadu nástrojů a komponent, díky kterým je poté implementace příjemnější. Většina obsahuje sady knihoven a šablon. Webové frameworky asistují režii na nižší úrovni, díky čemu se lze soustředit na věcnou stránku vyvíjené aplikace. Obecně lze konstatovat, že webové frameworky šetří čas i peníze při implementaci. Zpravidla se frameworky dělí na dva typy dle jejich určení na frontend a backend. [23]

3.6.1 Backend

Existuje široká škála backend frameworků pro vývoj webových aplikací. Liší se především způsoby práce s databázemi, rychlostí a učící křivkou. Pro účely webové aplikace byly přednostně zvažovány frameworky postavené na jazycích Python a PHP.

Django

Django³ je velmi rozšířený vysokoúrovňový webový framework založený na programovacím jazyku Python. Byl vytvořen pro zjednodušení vývoje webů, programátor díky němu nemusí znovu vymýšlet již existující řešení. Django podporuje využití vestavěné databázové knihovny SQLite3. Ta se v rámci frameworku využívá pro menší projekty a simulovaná data. Data z knihovny se v aplikacích využívají převážně pouze pro čtení, zároveň vyžadují menší instalační prostor. Výhodou je také takřka nulová konfigurace, není tedy nutné provádět složité instalace.

³<https://www.djangoproject.com/>

Framework je založen na systému MVC (angl. *Model-View-Controller*), což zaručuje mnohem rychlejší odezvu než jiné frameworky. I proto je vhodný zejména pro větší projekty, kde dokáže naplno prodat svůj potenciál. Zaměřuje se také na bezpečnost, a to zejména svými zabudovanými funkcemi *Cross-site request* a *SQL Injections*. Díky tomu v něm lze pracovat i s databázemi obsahující citlivá data. [17]

Flask

Flask⁴ je stejně jako Django taktéž založený na jazyce Python, narozdíl od něj je však tento backend framework jednodušší, a často se tak využívá v menších webových aplikacích. Byl vytvořen v roce 2010, tedy pět let po vzniku Django, a mohl se tak poučit z kritických reakcí okolo jeho komunity. Flask je s ohledem na jeho jednoduchost doporučovaný začátečníkům. Používá se často na menší a střední aplikace, kterými jsou například blogy, e-shopy a aplikace pro správu obsahů. Velkou výhodou Flasku je využití šablon a knihoven, kterým je například oblíbený jinja2⁵ pro vytváření šablon stránek. Díky této knihovně lze vytvořit dynamické HTML šablony doplněné o podmínky, proměnné a cykly.

PHP

Jazyk PHP⁶ je jedním z nejpoužívanějších pro vývoj backendu webových aplikací. PHP poskytuje mnoho frameworků, mezi nejznámější se řadí Laravel a Symfony. Oproti jazyku Python je PHP často rychlejší, hodí se tedy především pro aplikace s vysokým objemem datového provozu. Vývojáři volí PHP také při nutnosti využití více databází, jejich integrace je totiž jednodušší než ve frameworkích jazyku Python.

Laravel⁷ obsahuje mnoho předdefinovaných funkcí a knihoven. Mezi ně patří také ORM (angl. *Object Relational Mapping*), jenž zaručuje snadnou a rychlou práci s databází. Poskytuje databázovou abstrakci a stará se o dotazování na nová data. Dá se bez problému kombinovat s MySQL, Postgres, SQLite a další databázemi. Kromě toho cílí také na jednoduchou a bezpečnou autentizaci.

3.6.2 Frontend

Frontendové části webových aplikací tvoří především kombinace HTML, CSS a JavaScriptu. Knihoven JavaScriptu je na trhu nepřeberné množství, mezi nejpoužívanější patří React, Vue a Angular. Zjednodušení pro tvorbu CSS přináší knihovny s již hotovými komponentami. Zpřehlednění struktury CSS zajišťuje preprocesor SASS.

- **HTML** (angl. *HyperText Markup Language*) je standardní značkovací jazyk, jenž definuje strukturu webových stránek. Převládá v něm obecný text, který je doplněn elementy, což jsou úseky dokumentu vymezené tagy. [2]
- **CSS** (angl. *Cascading Style Sheets*) definuje vzhled webových stránek. Využívá se v kombinaci s HTML, na jehož strukturu se odkazuje pomocí selektorů. [12]
- **SASS** (angl. *Syntactically Awesome Style Sheets*) je vylepšení jazyka CSS. Zejména redukuje opakování stejných CSS bloků a díky tomu zpřehledňuje strukturu. Kód v jazyce SASS je nutné pomocí preprocesoru transpilovat na standard CSS. [26]

⁴<https://www.fullstackpython.com/flask.html>

⁵<https://www.fullstackpython.com/jinja2.html>

⁶<https://hackr.io/blog/python-vs-php>

⁷<https://laravel.com/>

- **JavaScript** je volně typovaný skriptovací jazyk s možností objektového programování, který se používá ve webových prohlížečích. Jádro jazyka je rozšířeno o objekty, které pomocí skriptů komunikují s uživatelem a mění obsah HTML dokumentu. JavaScript se stará o klientskou část webových aplikací. [9]

React

React je open-source knihovna jazyka JavaScript, jež pomáhá uživateli s vývojem front-endu webových aplikací. Původně byla vytvořena vývojáři Facebooku pro větší efektivitu firemní aplikace. Postupem času získal React obrovskou popularitu mezi programátory a podle statistik webu StackOverflow z roku 2021 se jednalo o vůbec nejpoužívanější webový framework na světě. [27]

Pro svou vnitřní strukturu využívá syntaktické rozšíření JSX (angl. *JavaScript XML*). Toto rozšíření umožňuje definovat v knihovně React elementy, které jsou velmi podobné syntaxi jazyka HTML. Tento přístup zjednodušuje vývojářům vytváření komponent, což jsou základní stavební bloky Reactu. Pro React je typická stromová struktura webové stránky označovaná jako virtuální DOM (angl. *Document Object Model*). Jakákoliv část HTML kódu je přidána jako jeden z uzlů této struktury. Právě k této struktuře lze pomocí skriptovacího jazyka JavaScript, na kterém je React postaven, jednoduše přistupovat. [1]

I díky aktivní podpoře komunity a vysoké flexibilitě se React řadí mezi nejvyužívanější front-end knihovny současnosti.

```

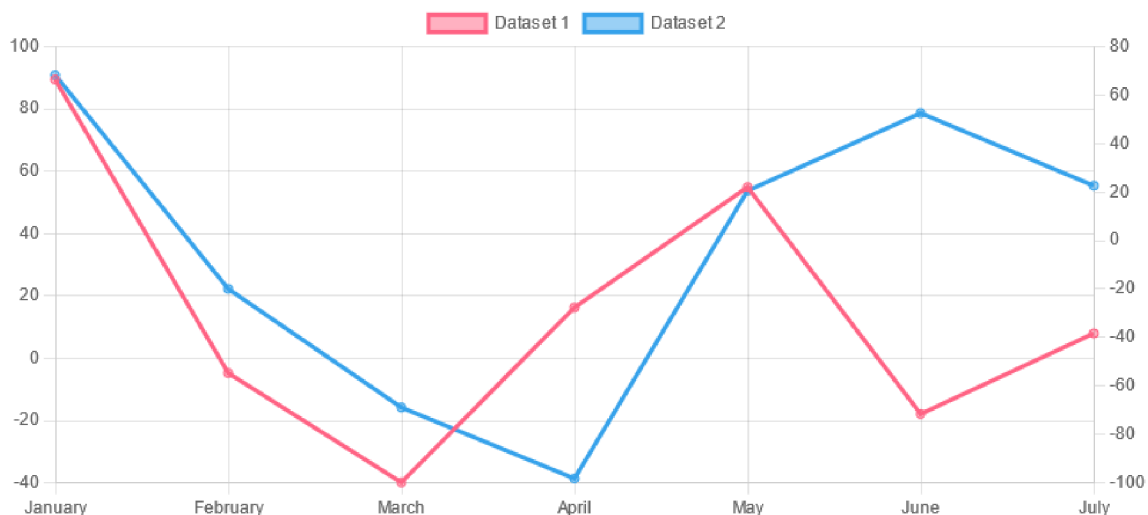
1   import React from 'react';
2
3   function MyComponent(props) {
4     return (
5       <div>
6         <h1>{props.title}</h1>
7         <p>{props.description}</p>
8       </div>
9     );
10  }
11
12  export default MyComponent;
```

Výpis 3.1: Vytvoření komponenty MyComponent v jazyku JavaScript za pomoci knihovny React. Komponenta bere jeden parametr `props`. Skládá se z kořenového tagu `div`, kde se nachází titulek a odstavec.

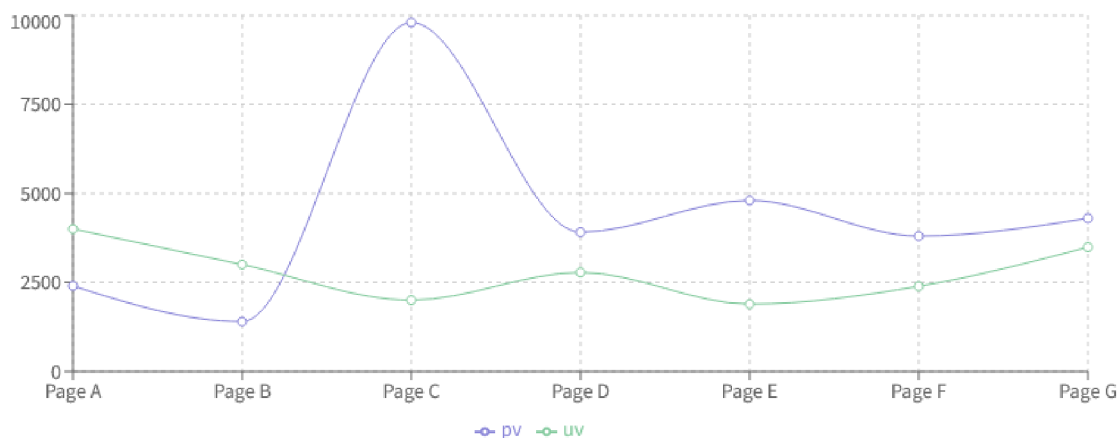
React obsahuje několik volně dostupných knihoven pro tvorbu grafů. Hojně využívanou je knihovna `Chart.js`⁸, která pro jazyk JavaScript poskytuje širokou škálu interaktivních grafů. `Chart.js` poskytuje také řadu možností nastavení, jako jsou různé barvy, popisky os a škálování. `React-Chartjs-2`⁹ je knihovna pro vytváření grafů a diagramů v React aplikacích pomocí `Chart.js`. `React-Chartjs-2` umožňuje snadné propojení s daty z React proměnných.

⁸<https://www.chartjs.org/>

⁹<https://react-chartjs-2.js.org/>



Obrázek 3.9: Chart.js umožňuje vykreslování více hodnot v jednom grafu. Součástí je legenda, která je modifikovatelná. Programátor má velmi pestrou škálu úprav grafů, například zde je liniový graf doplněn o dvě svislé osy s různými popisky.



Obrázek 3.10: Alternativou pro tvorbu interaktivních grafů je knihovna Recharts¹¹. Ta zajišťuje podobnou funkcionalitu jako Chart.js, navíc podporuje vyšší interaktivitu v grafu, jako je přibližování a posouvání. Recharts se zaměřuje i na zobrazení pro mobilní zařízení. Nevýhodou je výrazně nižší podpora komunity.

¹¹<https://recharts.org/en-US/>

Vue

Vue¹⁰ je open-source framework pro jazyk JavaScript, který slouží k všestrannému vývoji uživatelských rozhraní. Je postaven na jazycích HTML, CSS a JavaScript a podobně jako React poskytuje programovací model založený na komponentách. Vue stejně jako React využívá virtuální DOM. Změny se neprovádí přímo v DOM, místo něj se vytvoří jeho replika ve formě datových struktur jazyka JavaScript. Změny se nejprve provedou v datových strukturách a až finální změny se aktualizují do skutečného DOM. Tento přístup je vhodný z hlediska optimalizace výkonu, změny se provádí rychleji za menší datovou cenu. Vue se vyznačuje se svou jednoduchostí a malou velikostí, to ovšem může přinášet problémy ve větších projektech.

```
1   <template>
2     <div>
3       <h1>{{ title }}</h1>
4       <p>{{ description }}</p>
5     </div>
6   </template>
7
8   <script>
9     export default {
10      name: 'MyComponent',
11      props: {
12        title: String,
13        description: String
14      }
15    }
16  </script>
```

Výpis 3.2: Úsek HTML kódu s komponentou MyComponent vytvořenou pomocí frameworku Vue. Komponenta obsahuje dvě vlastnosti značící titulek a popisek. Tyto vlastnosti jsou předávány do rodičovské komponenty.

Angular

Angular je open-source framework pro vývoj webových aplikací vyvíjený společností Google. Využívá jazyka TypeScript, který rozšiřuje jazyk JavaScript o podporu modularity a brzkého odhalení chyb i mimo dobu běhu programu. Největší výhodou TypeScriptu je kompilátor, který vyhledává chyby. Zobrazení chyb v reálném čase je výhodné především pro refaktorizaci kódu. Komunita kolem jazyka JavaScript a jeho frameworků je však výrazně větší. [22]

Angular má strmou křivku učení, začátečníkům může trvat delší dobu pochopit jeho základní fundamenty. S tím souvisí také jazyk TypeScript, jehož odlišnosti vzhledem k JavaScriptu je nutné prostudovat. Kvůli své těžkotonážní architektuře není vhodný pro malé projekty, je určen spíše pro komplexnější systémy. Zároveň tento framework není vhodný pro statické weby kvůli problémům s přístupností. [25]

¹⁰<https://vuejs.org/>

3.6.3 Shrnutí

Analýza dostupných technologií pokryla jen několik dostupných webových frameworků a knihoven. Průzkum byl zaměřen na nejpoblárnější technologie mezi komunitami a nejvhodnější prostředky pro vytvářenou aplikaci.

Z prozkoumaných technologií byl pro vytvoření serverové části webové aplikace vybrán framework Django. Hlavní výhodou oproti PHP frameworkům byla zkušenost s jazykem Python a také větší komunita okolo tohoto jazyka. Klientskou část webové aplikace obstará knihovna React. Do budoucna jsou aplikace vytvořené v Reactu nejvíce udržitelné, a to především díky výrazné podpoře komunity a progresivnímu vývoji knihovny. React zároveň obsahuje balíčky pro tvorbu interaktivních grafů, které se pro realizaci statistických sekcí z návrhu hodí.

Technologie	Výhody	Nevýhody
Django	rychlost, strukturovatelnost, bezpečnost, komunita	učící křivka, práce s komplexními databázemi
Flask	jednoduchost, komunita	limitovaná funkcionalita
Laravel	práce s databázemi, autentizace	učící křivka, nižší rychlost
React	učící křivka, knihovny a komponenty	instalace mnoha balíčků
Vue	učící křivka, jednoduchost, rychlost	menší komunita, limitovaná funkcionalita
Angular	komplexnost, robustnost	učící křivka, nižší rychlost, TypeScript

Tabulka 3.3: Flask neoslovil kvůli své omezené funkcionalitě. Vue byl zavržen kvůli nižší udržitelnosti a menší podpoře komunity. Angular je pro projekt příliš robustní a nemusel by zaručovat adekvátní rychlost. Zároveň je i díky předchozím zkušenostem preferovaným jazykem JavaScript oproti TypeScriptu.

Kapitola 4

Implementace

Architektura webové aplikace funguje na modelu klient-server. V této architektuře je server specializovaný na databázové dotazy, které se po síti přenášejí ze strany klienta. Server vytvoří odpověď a posílá ji zpět na klienta. [3]

Většinou část komunikace mezi klientem a serverem tvoří asynchronní požadavky na API¹ (angl. *Application Programming Interface*). To se děje při každé změně v nastavení, selekci dat v grafu či statistikách, získání nového úspěchu a také pro aktualizaci dat v přehledové sekci podle pravidelného časového intervalu. Komunikace směrem k serveru probíhá také při žádostech o poskytnutí jiné webové stránky po změně URL, například stiskem tlačítka v hlavním menu.

Struktura projektu je rozdělena na tři části, ze kterých jedna zajišťuje backend a druhá frontend. Třetí část je kořenová složka projektu, která sdružuje SQLite3 databázi a skripty pro spuštění funkcí na generování simulovaných dat.

První aplikace `glasshouse` se stará o veškerou práci s databází, je tedy serverovou částí aplikace. Obsahuje databázové modely, funkce pro generování simulovaných dat a všechny pohledy generující API, na které se pro získání dat dotazuje klientská část aplikace. Figuruje zde také nastavení administrátorského pohledu Django.

Klientská část je v aplikaci `glasshouseapp`. Ta obsahuje veškeré komponenty vytvořené v Reactu a stylování pomocí kaskádových stylů.

- `glasshouse/` - databázové modely, mapování URL, tvorba API, tvorba simulovaných dat, nastavení projektu, autentizace, administrátorský pohled
- `glasshouseapp/` - prezentační prvky, komponenty, stylování, obrázky
- `/` - databáze SQLite3, spouštěče generátorů simulovaných dat

4.1 Serverová část

Veškerá část aplikace ze strany serveru je v Django aplikaci `glasshouse`. Tvoří ji zejména databázové modely, z jejichž dat jsou vytvořeny koncové body API pro přístup z klientské strany. Součástí jsou také skripty pro generování simulovaných dat.

¹rozhraní mezi různými softwarovými aplikacemi, které umožňuje výměnu dat

4.1.1 Databázové modely

V databázi jsou celkově čtyři druhy záznamů. Všechny jsou uloženy v databázi SQLite3, která je součástí projektu ve frameworku Django. Pro každý záznam v modelech je generováno unikátní číslo `id`, přes tento primární klíč jsou jednotlivé záznamy odkazovány. Při generování testovacích dat se primární klíč stanovuje automaticky počínaje číslem 1.

Pro veškerou funkcionalitu webové aplikace bylo nutné vytvořit záznamy s historií naměřených dat ze senzorů. Tyto záznamy jsou v modelu nazvaném `UserData` s následujícími atributy:

- **timestamp** - časové razítko, tvořeno pro záznamy v intervalu jedné hodiny
- **inside_temp** - vnitřní teplota vzduchu [°C]
- **inside_humid** - vnitřní vlhkost vzduchu [%]
- **outside_temp** - venkovní teplota vzduchu [°C]
- **outside_humid** - venkovní vlhkost vzduchu [%]
- **soil_humid_1** - vlhkost půdy 1. zavlažovacího okruhu [%]
- **soil_humid_2** - vlhkost půdy 2. zavlažovacího okruhu [%]
- **roof_window** - stav střešního okna [ON|OFF]
- **heating** - stav vyhřívání [ON|OFF]
- **fan** - stav ventilátoru [ON|OFF]
- **watering_1** - stav zavlažování 1. okruhu [ON|OFF]
- **watering_2** - stav zavlažování 2. okruhu [ON|OFF]
- **sprinkler** - stav rozprašovače [ON|OFF]

Pro záznamy s teplotami a vlhkostmi je využito číslo s plovoucí desetinnou čárkou zaokrouhlené na dvě desetinná místa. Stavby se do záznamů ukládají jako celá čísla, kde nula značí vypnuto a 100 zapnuto. Tento způsob uložení dat byl zvolen díky jejich jednoduššímu zpracování pro graf.

Dalším modelem ve struktuře databáze je `StatsData`. Do něj se ukládají data o spotřebě vody a elektrických spotřebičů. Jsou zde také záznamy o celkové době provozu aktuátorů a zavlažovacích okruhů. Model má kromě svého primárního klíče `id` tyto atributy:

- **timestamp** - časové razítko, tvořeno pro záznamy v intervalu jednoho dne
- **fan_day** - doba provozu ventilátoru [min]
- **heating_day** - doba provozu vyhřívání [min]
- **watering_1_day** - doba provozu 1. okruhu zavlažování [min]
- **watering_2_day** - doba provozu 2. okruhu zavlažování [min]
- **lights_day** - doba provozu osvětlení [min]

- **lights_energy_day** - spotřeba elektrické energie z osvětlení [kWh]
- **electrical_energy_day** - celková spotřeba elektrické energie [kWh]
- **water_consumption_day** - celkový průtok vody [l]

Data o celkové době spuštění spotřebičů se do databáze ukládají v minutách. Pro přehlednější zobrazení se poté v aplikaci převedou do formátu dnů, hodin a minut.

Uživatelské nastavení podmínek má další model. Nový záznam se vytvoří při každé změně v postranním menu s nastavením. Díky tomu lze uchovat historii změn podmínek v chytrém skleníku. Model `OverviewSettings` má následující atributy:

- **timestamp** - časové razítko, zaznamená se při každém novém PUT požadavku
- **temperature_low** - dolní hranice teploty vzduchu [°C]
- **temperature_high** - horní hranice teploty vzduchu [°C]
- **humidity_low** - dolní hranice vlhkosti vzduchu [%]
- **humidity_high** - horní hranice vlhkosti vzduchu [%]
- **heating_set** - stav vyhřívání [on|off|auto]
- **fan_set** - stav ventilátoru [on|off|auto]
- **roof_window_set** - stav střešního okna [on|off|auto]
- **watering1_set** - stav 1. zavlažovacího okruhu [on|off|auto]
- **watering1_low** - dolní hranice 1. zavlažovacího okruhu [%]
- **watering1_high** - horní hranice 1. zavlažovacího okruhu [%]
- **watering2_set** - stav 2. zavlažovacího okruhu [on|off|auto]
- **watering2_low** - dolní hranice 2. zavlažovacího okruhu [%]
- **watering2_high** - horní hranice 2. zavlažovacího okruhu [%]
- **sprinkler_set** - stav stropního rozprašovače [on|off|auto]
- **camera_interval** - interval snímání dohledové kamery [hod]

Zde je stav spotřebičů zanesen přímo podle aktuálního stavu třístavového tlačítka. Veškeré změny nastavení podmínek se provedou až po stisku potvrzovacího tlačítka.

Pro systém gamifikace byl vytvořen model `Achievements`. Zde jsou uloženy veškeré úspěchy, které může uživatel získat. Tato data jsou již předem definovaná. Model obsahuje následující atributy:

- **title** - titulek úspěchu
- **task** - úkol pro získání úspěchu
- **state** - úroveň úspěchu, určuje počet pohárů [1|2|3]

- **achieved** - určuje zdali byl úspěch splněn [0|1]
- **achieve_date** - datum splnění úkolu

Po získání úspěchu se změní pouze atribut **achieved**, který nabývá stavu 1 při splněním úspěchu. Na začátku je každý úspěch nesplněný, tedy se stavem 0. Zároveň se do atributu **achieve_date** zanesou aktuální datum. Atribut **state** udává úroveň úspěchu danou třemi číslicemi. První úroveň je nesnazší na získání a třetí, poslední úroveň, je naopak nejtěžší získat.

4.1.2 Koncové body API

Pro přístup k datům z databáze byla v Django vytvořena sada API endpointů². API endpointy použité v aplikaci lze rozdělit do dvou kategorií. První data pouze získává, druhá kategorie s daty manipuluje pomocí aktualizací či nových záznamů. Každý koncový bod je definován svou URL adresou, ke které se přistupuje pomocí požadavků HTTP³ (angl. *Hypertext Transfer Protocol*). Všechny koncové body mají stejnou URL cestu, nachází se v adresáři `api/`.

URL	Metoda	Funkce
<code>api/get-data/</code>	GET	Vrátí všechny záznamy o naměřených datech ze senzorů
<code>api/get-last-object/</code>	GET	Vrátí poslední záznam z naměřených dat ze senzorů
<code>api/get-stats-data/</code>	GET	Vrátí všechny záznamy statistik spotřebičů
<code>api/get-settings/</code>	GET	Vrátí aktuální nastavení podmínek
<code>api/get-achievements/</code>	GET	Vrátí záznamy o všech úspěších
<code>api/update-settings/</code>	POST	Vytvoří nový záznam po aktualizaci v nastavení podmínek teploty, vlhkosti a spotřebičů
<code>api/update-water-settings/</code>	POST	Vytvoří nový záznam po aktualizaci v nastavení podmínek zavlažování
<code>api/update-camera-settings/</code>	POST	Vytvoří nový záznam po aktualizaci v nastavení kamery
<code>api/update-achievement/<int:id>/</code>	PUT	Aktualizuje záznam úspěchu s identifikátorem <code>id</code>

Tabulka 4.1: Koncové body API.

Každé API s metodou GET vygeneruje serializovaný řetězec JSON⁴ (angl. *JavaScript Object Notation*) s požadovanými daty. Pro realizaci je využita funkce `JsonResponse` z běžně dostupného modulu frameworku Django `django.http`. Tato funkce umožňuje vytvářet HTTP odpovědi ve formátu JSON. Funkce `JsonResponse` si na prvním argumentu přebírá

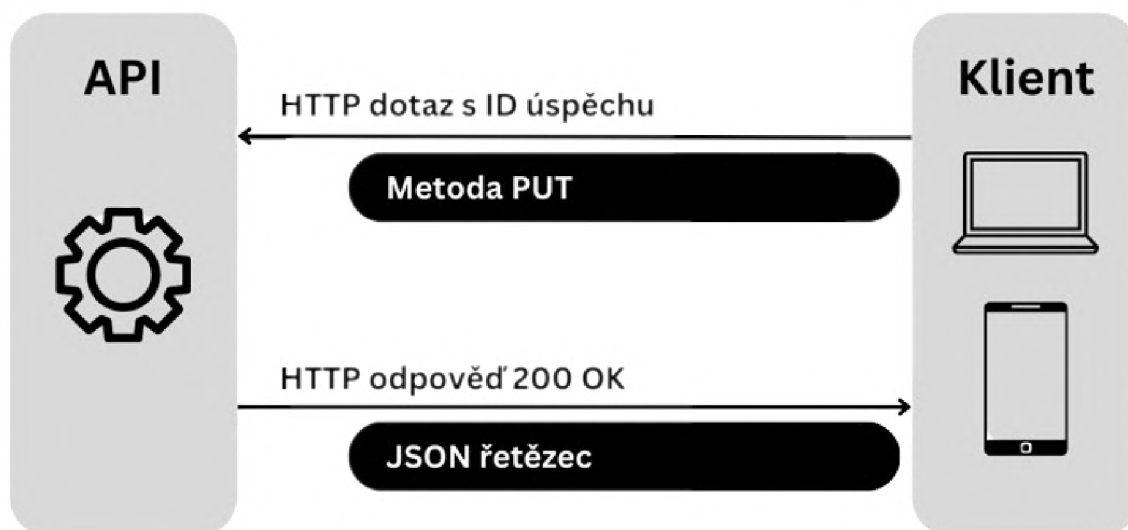
²koncový bod s unikátní URL, který umožňuje přístup k datům

³protokol pro komunikaci mezi webovými prohlížeči a webovými servery

⁴formát JSON umožňuje reprezentovat objekty jako čísla, řetězce a pole v hierarchické struktuře

předem upravená data ve formátu klíč-hodnota. Odpověď si přebere dotazovaná klientská část aplikace a dále s ní pracuje.

Zpracování dotazů POST funguje pro každé API stejně. Kvůli zachování historie uživatelského nastavení se při každé změně zaneše nový záznam. Pokud uživatel změni například pouze horní hranici teploty vzduchu, tak se po potvrzení změny vytvoří nový záznam se zadanou hodnotou horní hranice teploty vzduchu a zbytek dat se převezme z posledního přidaného záznamu. Záznamy jsou unikátní díky identifikátoru `id` a časovému razítku `timestamp`.



Obrázek 4.1: Aktualizace stavu úspěchů je realizovaná pomocí PUT požadavku z klientské strany aplikace. Každý úspěch má jednoznačný identifikátor, který se přenáší v požadavku na API. Při zadání neexistujícího `id` se navrátí textová odpověď se stavovým kódem 404.

Pro účely vývoje aplikace na lokálním serveru bylo nutné opatřit funkce zpracovávající HTTP metody o dekorátor `@csrf_exempt`. Ten slouží k vypnutí ochrany proti CSRF útokům (angl. *Cross-Site Request Forgery*), které využívají aktivní relaci přihlášeného uživatele a pokouší se vykonat na stránce nějakou akci, například změnu hesla. Pohledy totiž přijímají data z externího systému aplikace React, který nedokáže poskytnout očekávané CSRF tokeny.

4.1.3 Testovací data

Aplikace nepracuje s reálnou datovou sadou, proto bylo nutné vytvořit simulovaná data. Generátory simulovaných dat se nachází v souboru `utils.py`. Funkce z tohoto souboru jsou volány z generovacích skriptů v kořenovém adresáři projektu.

O veškeré generování dat pro simulaci podmínek v chytrém skleníku se stará funkce `insertRandomData`, která přebírá parametr `days` značící posun dní do minulosti, pro které se budou tvořit data. Generování dat skončí aktuálním časovým razítkem. Nejprve se vygenerují záznamy pro model `UserData`, které od sebe dělí hodina času. Hodnoty pro teploty a vlhkosti jsou při každé iteraci stanoveny generátorem pseudonáhodných čísel `random` vestavěného modulu jazyka Python. Data jsou vybírána z intervalu hodnot, ve kterých se v reálném prostředí nejčastěji pohybují. Pro spotřebiče, střešní okno a zavlažovací okruhy

je vybrána pomocí metody `random.choice` jedna z hodnot zapnuto-vypnuto. Ty jsou pro zjednodušení práce s grafy realizovány čísly 0 pro vypnuto a 100 pro zapnuto.

Následně dojde ke generování záznamů pro model `StatsData`. Tyto záznamy jsou tvořeny pro jeden celý den, tedy bez razítek hodin a minut. Data o spotřebách elektrické energie a vody jsou generována pomocí metody `random.uniform` v rozmezí hodnot blízkému reálnému skleníku menších rozměrů. Údaje o době spuštění spotřebičů se pohybují v rozmezí 0 až 24 hodin.

Nakonec dojde ke generování záznamů pro uživatelské nastavení podmínek v modelu `OverviewSettings`. Zde se záznamy vytvoří taktéž pro každý jeden den, ovšem časové razítko obsahuje i hodiny a minuty. Pro táhla se vygenerují údaje z nejčastěji nastavovaných hodnot z reálného skleníku. Pro spotřebiče a střešní okno se náhodně vybere jedna z hodnot ON, OFF, AUTO.

Soubor obsahuje také funkci `deleteData`, která vymaže všechny záznamy z databáze. Ponechá pouze záznamy o úspěších, které nejsou generované. Nakonec zde figuruje funkce `resetAchievements`, jež po zavolání nastaví stav všech úspěchů na 0 (nesplněno) a vymaže všechna časová razítka o splnění. Obě funkce sloužily při vývoji k testování aplikace a ke snadné manipulaci s daty.

4.1.4 Autentizace

Autentizační proces je implementován pomocí JWT (angl. *JSON Web Token*), což je standard pro výměnu informací jako JSON objekt mezi stranami, například mezi klientem a serverem. Přístupový token se využívá pro ověření identity uživatele. Využívá se také obnovovací token, který je dlouhodobější a slouží k obnovení přístupového tokenu bez nutnosti nového přihlášení z pohledu uživatele. Po přihlášení je součástí každého požadavku na server také vygenerovaný přístupový token, řetězec písmen a čísel, který uživatele opravňuje k využití chráněných zdrojů aplikace. Vygenerovaný řetězec je navíc opatřen elektronickým podpisem pomocí hashovacího algoritmu. V aplikaci je pro tento účel využitý algoritmus HS256. Tokenům lze nastavit různou životnost, tedy dobu po kterou zůstávají platné.

Implementace přihlašovacího procesu je realizována za pomoci volně dostupné knihovny `Django Simple JWT`. Ta poskytl jednoduchý způsob pro vytvoření a ověřování JWT uživatele. Z knihovny je využita třída `TokenObtainPairView`, jež vytvoří přístupový token pro uživatele po přihlášení. Tento token se poté využívá v dalších požadavcích na API. Knihovna poskytuje také obnovovací token. Knihovna nabízí i konfigurační možnosti, ve kterých byla nastavena doba přístupového tokenu na 30 minut, obnovovací token je v aplikaci platný po dobu jednoho dne.

Data o uživateli aplikace sdružuje model `UserLogin`, jenž dědí z abstraktní třídy `AbstractUser`. Dědění z této třídy zajišťuje pro model základní funkce a atributy, kterými jsou i potřebné uživatelské jméno a heslo. V případě potřeby se dají tyto atributy rozšířit.

4.2 Klientská část

O stěžejní klientskou část aplikace se stará frontend knihovna React. Veškeré soubory k prezentační části lze nalézt v aplikaci `glasshouseapp`. Nejvyšší úroveň hierarchie komponent tvoří základní kámen aplikace `App`. Tato kořenová komponenta je v Reactu kostrou aplikace a obsahuje další komponenty, které tvoří celkové uživatelské rozhraní.

V `App` je implementována logika kolem systému gamifikace. Ta totiž prochází všemi částmi webové aplikace, a je tak řízena přímo z kořenové komponenty. Při splnění některého z úspěchů v dílčích komponentách, se v `App` provede dotazování na API pro aktualizaci splněného úspěchu.

Komponenta `App` se také stará o směrování webových adres. To je implementováno pomocí volně dostupné komponenty `BrowserRouter`, ve které lze vytvořit kompletní adresové směrování. Vybraná adresa je v navigaci automaticky označena třídou jako aktivní, což ulehčuje práci s vizuálním odlišením aktivních adres v menu.

4.2.1 Přihlášení uživatele

Přihlášení uživatele do aplikace je provedeno pomocí loginu a hesla. Jelikož webová aplikace patří k fyzickému produktu, chytrému skleníku, nelze si samovolně vytvářet nové účty. Přihlašovací údaje získá uživatel zakoupením fyzického chytrého skleníku. Po vyplnění správných přihlašovacích údajů a následného potvrzení se vytvoří požadavek na API pro získání přístupového tokenu. Při nesprávně zadaných údajích se uživateli zobrazí textová zpráva o této chybě.

4.2.2 Přehledová obrazovka

Po přihlášení do webové aplikace se uživatel dostane na kořenovou adresu skrývající přehledovou sekci. Hlavní částí je grafická vizualizace obecného skleníku doplněná o sekce s měřenými daty. Pro přehlednost jsou vytvořeny čtyři sekce v každém rohu modelu. V rozích nalevo figurují dva elementy s aktuálními hodnotami vnitřního a venkovního měření s teplotou a vlhkostí vzduchu. V pravých rozích jsou poté uvedeny aktuální stavy spotřebičů, střešního okna a zavlažovacích okruhů i s rozprašovačem. Grafický model skleníku doplňují elementy zobrazující aktuální hodnoty vlhkosti půdy.

Při kliknutí na elementy v postranních rozích se uživatel dostane do postranní nabídky nastavení podmínek ve skleníku. Tato akce není na elementu zobrazujícím venkovní měření, jelikož zde nelze nic nastavovat. Přehledová sekce je implementována v komponentě `Overview`, která jako všechny komponenty kromě kořenové `App` přijímá parametry `settingsAchievements` a `achievementDone`, které se starají o vyhodnocení úspěchů a následný požadavek o aktualizaci daného úspěchu.

Měřená data se asynchronně aktualizují podle časového intervalu. V simulovaných datech jsou zaznamenány údaje každou hodinu, proto je v řešení nastaven interval na 10 minut. Při zvýšení frekvence logování dat je nutné tento interval snížit.



(a) Venkovní měření obsahuje teplotu a relativní vlhkost vzduchu.

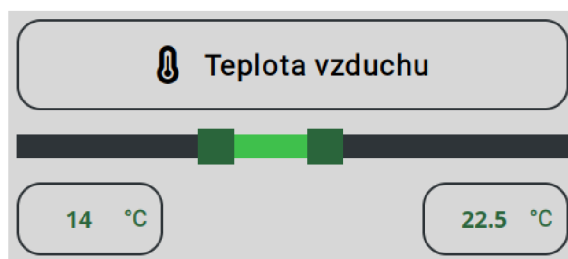


(b) Ventily pro zavlažování nabývají stavu ON nebo OFF.

Obrázek 4.2: Grafické elementy pro aktuálně měřené hodnoty zobrazené na vizualizaci skleníku v hlavní přehledové sekci.

4.2.3 Řízení podmínek

Řízení podmínek v inteligentním skleníku se provádí v rozbalovací postranní nabídce. Do té se uživatel dostane po kliknutí na rohové elementy s aktuálně měřenými daty v přehledové sekci. Pro nastavení vnitřní vlhkosti a teploty vzduchu je nutné kliknout na element v levém horním rohu zobrazující vnitřní měření.



Obrázek 4.3: První možností řízení teploty a vlhkosti vzduchu je selekce intervalu hodnot pomocí táhla se dvěma koncovými body. Krok pro teplotu je nastaven na 0,5 °C, pro vlhkost vzduchu na 1 %. Druhou možností výběru je pole, které přijme jakoukoliv číselnou hodnotu. Vstup je ošetřen proti nevalidním znakům. Zároveň do levého pole značícího dolní hranici nelze uvést vyšší hodnotu, než je v pravém poli, jenž udává horní hranici. Změna v poli se automaticky promítá do stavu táhla a naopak, a to již v průběhu selekce hodnot.

Do stejné sekce s nastavením se uživatel dostane i po kliknutí na element v pravém horním rohu, jenž zobrazuje hodnoty střešního okna, ventilátoru a vyhřívání. Pro snížení počtu různých nastavovacích nabídek je nastavení těchto částí chytrého skleníku sdruženo s řízením podmínek vzduchu. Nastavení aktuátorů se provádí pomocí třístavových tlačítek.

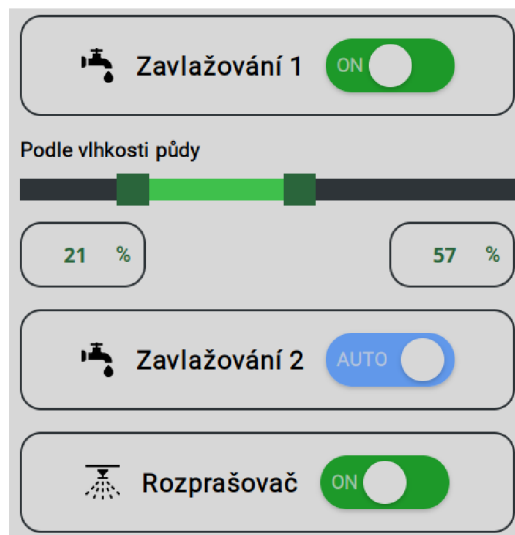
Nabídka nastavení je implementována v komponentě `Menu_settings`. Ta přebírá kromě parametrů pro práci s úspěchy také dva parametry o stavu postranního menu. První z nich

`menuOpen` nabývá jednoho ze dvou stavů `True` nebo `False`, a uchovává tak v sobě informaci o otevření postranního menu. Druhý parametr je funkce `setMenuOpen`, ten zajišťuje aktualizaci stavu `menuOpen` při otevření či zavření postranní nabídky s nastavením. Změny podmínek je nutné potvrdit pomocí tlačítka v dolní části menu. Po stisku tlačítka pro potvrzení změn dojde k POST požadavku na API a vytvoří se nový záznam s nastavením podmínek. Při úspěšném vytvoření záznamu dojde k zavření postranní nabídky a uživateli se po dobu pěti sekund ukáže potvrzovací zpráva o zdárném uložení nových hodnot. Je zde také možnost pro zavření postranní nabídky bez uložení změn.

Nabídka pro nastavení zavlažovacích okruhů a rozprašovače se obsahem od výše popsaného menu liší. Zavlažovací okruhy se nejprve nastavují taktéž pomocí třístavového tlačítka na jeden ze stavů `ON`, `OFF` a `AUTO`. Při stavu `ON` se navíc zobrazí táhlo a pole pro výběr hodnot. Uživatel zde nastavuje rozmezí vlhkosti půdy. Chytrý skleník poté udržuje vlhkost půdy ve stanoveném intervalu na základě naměřených dat z půdních senzorů. Stropní rozprašovač se nastavuje stejným třístavovým přepínačem. Sekce s postranní navigací pro zavlažování je implementována v komponentě `Water_settings`.



(a) Nastavení zařízení a střešního okna se provádí pomocí třístavových přepínačů.



(b) Zavlažování se nastavuje pomocí třístavových přepínačů. Při zapnutém stavu lze navíc pomocí posuvníku vybrat rozmezí hodnot, ve kterých se bude udržovat vlhkost půdy.

Obrázek 4.4: Způsoby nastavení podmínek v rozbalovacím menu aplikace.

4.2.4 Graf

Graf má v aplikaci samostatnou sekci se stejnojmenným názvem. Graf je vytvořen pomocí volně dostupné knihovny Chart.js. V aplikaci je implementovaný liniový graf, ve kterém lze zobrazit všechna měřená data naráz. Z knihovny Chart.js je využita její základní komponenta `Line`, do které jsou předána data pro zobrazení a také pro nastavení.



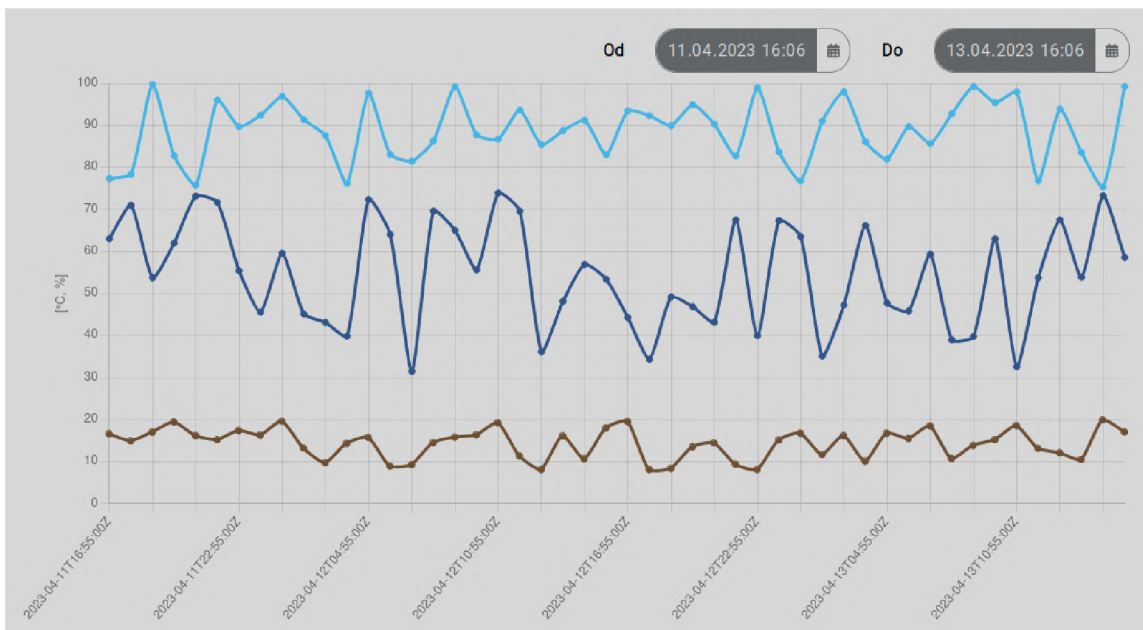
Obrázek 4.5: Automaticky generovaná legenda nesplňovala očekávání uživatelů z testování a nekorespondovala s návrhem. Proto byla pro selekci dat v grafu vytvořena nová interaktivní legenda. Výběr dat do grafu probíhá stiskem levého tlačítka myši na neaktivní popisek či obdélník v legendě. Na ukázce vytvořené legendy jsou v grafu aktivní čtyři hodnoty.

O viditelnost dat v grafu se stará vnitřní proměnná `isDatasetVisible`. Ta obsahuje v poli indexy všech datasetů, které mají být skryté. Na základě toho se pak při aktualizaci stavu vykreslí do grafu jen ty hodnoty, jejichž indexy v tomto poli nejsou. Změny se dějí po stisku na popisky v legendě, což vyvolá funkci, která předává index datasetu. Pokud proměnná `isDatasetVisible` index obsahuje, tedy vybraná data nejsou viditelná, tak se provede přidání indexu do pole a pro legendu na předaném indexu se nastaví aktivní stav. V legendě se aktivní data, která jsou zobrazená v grafu, vyznačují barevným obdélníkem a nepřeskrtnutým textem. Neaktivní data mají přeškrtnutý světlejší text a šedý obdélník.

V grafu je možné filtrovat data podle časového razítka záznamu. Datum a čas se vybírá pomocí interaktivního kalendáře. Po stisku ikony kalendáře se zobrazí kalendář se dny ve vybraném měsíci. V kalendáři lze snadno procházet mezi měsíci i roky. Dnešní datum je v kalendáři odlišeno barevně. Hodiny a minuty se v elementu zadávají ručně pomocí klávesnice. Datum i minutáž lze navíc měnit i pomocí kolečka myši.

Ve výchozím nastavení jsou všechny hodnoty grafu v neaktivním stavu. Horní hranice časového razítka je na začátku nastavena automaticky na aktuální datum a čas. Spodní hranice je ve výchozím stavu na pět dní starém datu. Data se v grafu mění již při selekci dat v kalendáři, pro jednoduchost zde není žádné potvrzovací tlačítko. Smazání výběru všech zobrazených dat v grafu je možné provést pomocí jednoho tlačítka.

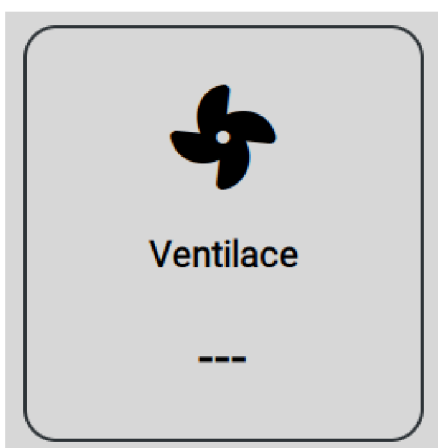
Data z vybraného časového období je možné exportovat do souboru v CSV formátu. O tento proces se stará volně dostupná komponenta `CSVLink`. Ve vybraném časovém intervalu z kalendářů se vždy exportují všechna měřená data, tedy i ta neaktivní v grafu.



Obrázek 4.6: Zobrazení třech hodnot v grafu s výběrem časového rozmezí v kalendářích za posledních 48 hodin měření.

4.2.5 Statistiky

Pro statistická data z provozu spotřebičů byla vytvořena samostatná sekce Statistiky. Její implementace je v komponentě **Stats**. V této sekci uživatel zjistí, jak dlouho jely ve vybraném časové úseku jednotlivé spotřebiče. Zároveň zde má možnost porovnat celkovou spotřebu elektrické energie. Vidí zde také statistiku spotřeby elektrické energie přímo pro osvětlení, které ve sklenících zabírá podstatnou část celkové denní spotřeby. Nechybí zde ani údaje o spotřebě vody.



(a) Zobrazení statistiky při výběru neexistujících dat.



(b) Spotřeba elektrické energie na osvětlení.

Obrázek 4.7: Statistika spotřeby energií a doby spuštění spotřebičů.

Statistická data jsou v simulovaných datech uvedena za jeden celý den. Proto lze v přiloženém kalendáři, vytvořeném stejně jako v grafu pomocí komponenty `DateTimePicker`, filtrovat data pouze za celé dny. Vyfiltrovaná data ze statistik je možné taktéž exportovat do CSV souboru.

V databázi jsou denní doby spuštění spotřebičů uloženy v celých minutách. Pro reprezentaci údajů v aplikaci bylo nutné vytvořit funkci, jež konvertuje minuty na formát dnů, hodin a minut pro snazší orientaci uživatele. Pokud je ve dvojici kalendářů vybráno takové rozmezí dat, pro které neexistuje v databázi žádný záznam, zobrazí se pro místo reálných hodnot pouze prázdný ukazatel.

4.2.6 Snímky dohledové kamery

Uživatel si v aplikaci může prohlížet snímky pořízené dohledovou kamerou ve skleníku. Lze mezi nimi procházet šipkami v dolní části obrazovky. Aktuálně vybranou fotografii je možné stáhnout pomocí ikony uložení.



Obrázek 4.8: Pro kameru je možné nastavovat interval snímání, ten lze nastavit na maximálně 24 hodin, minimálně pak na jednu hodinu. Pro výběr hodnoty je k dispozici táhlo a textové pole `input`. Při nastavení intervalu na nula hodin se snímání považuje za vypnuté.

4.2.7 Systém úspěchů

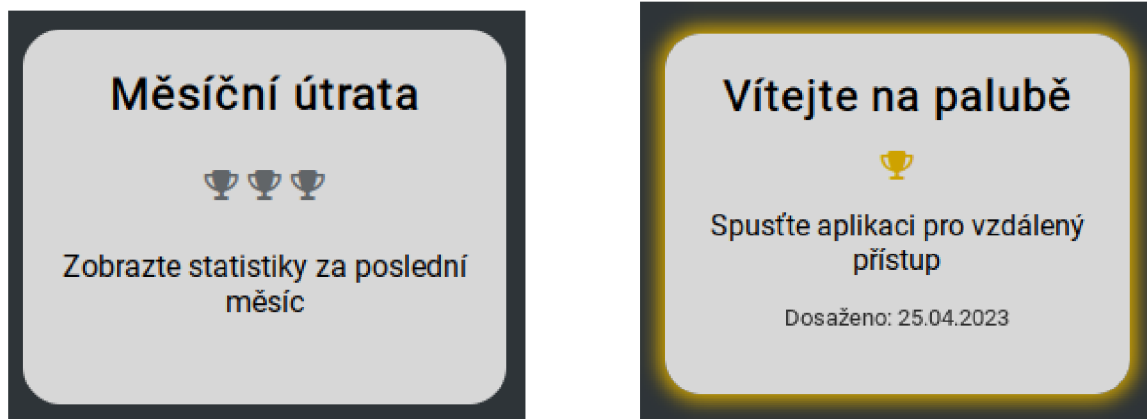
Systém úspěchů provází všechny části webové aplikace. Jeho centrálním bodem je ovšem kořenová komponenta `App`, ve které dochází k dotazování na API s úspěchy a také vyhodnocování nových úspěchů. Zároveň se z této komponenty zobrazí zpráva o splnění úkolu. V kořenové části aplikace je také možné získat úspěchy za splnění určitého počtu úkolů.

Při splnění některého z úkolů se z dílčích komponent se zavolá funkce `achievementDone`, jež bere jeden parametr. Tím je `id`, ve kterém se předává jednoznačný identifikátor úspěchu, který je nutné aktualizovat. Na začátku jsou všechny úspěchy ve výchozím stavu nula, jež značí, že daný úspěch ještě nebyl splněn. Při splnění přejde úspěch do splněného stavu. Každý úspěch má jinou úroveň, tudíž je za něj také jiné ohodnocení dané počtem pohárů. Úroveň úspěchu odráží jeho náročnost na splnění. Minimální úroveň je jeden pohár, maximální jsou poté poháry tři.

Při splnění úspěchu se do záznamu uloží také aktuální datum, aby uživatel věděl kdy daný úspěch splnil. Aktualizace úspěchů v databázi se provádí pomocí HTTP požadavku metodou `PUT`. Při úspěšné aktualizaci záznamu se zavolá funkce `setLastAchieved`, která nastavuje interní proměnnou vždy na identifikátor posledního splněného úspěchu. Nakonec se uživa-

teli v levém dolním rohu obrazovky ukáže zpráva o úspěšném splnění úkolu, jež obsahuje titulek a popis úkolu. Zprávu je možné zavřít po stisknutí myši na její obsah.

Všechny úspěchy jsou uvedeny v samostatné sekci z menu nazvané Úspěchy. Každý z úspěchů obsahuje titulek, poháry a popis úkolu. Dokončené a nedokončené úspěchy se odlišují barvou pohárů, stínem elementu a časovým razítkem dokončení úkolu.



(a) Nedokončený úkol s maximálním oceněním tří pohárů.

(b) Dokončený základní úkol, který je ohodnocen jedním pohárem.

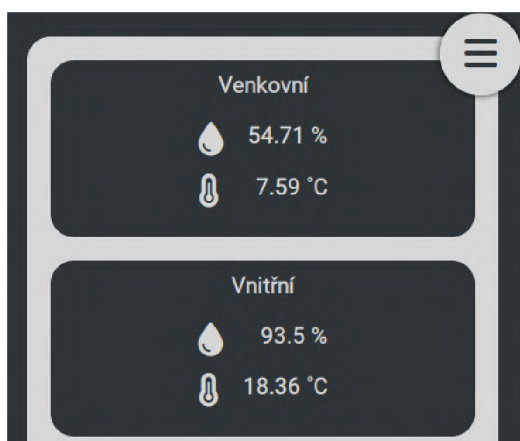
Obrázek 4.9: Grafická reprezentace úkolů v různých stavech.

4.2.8 Verze pro mobilní zařízení

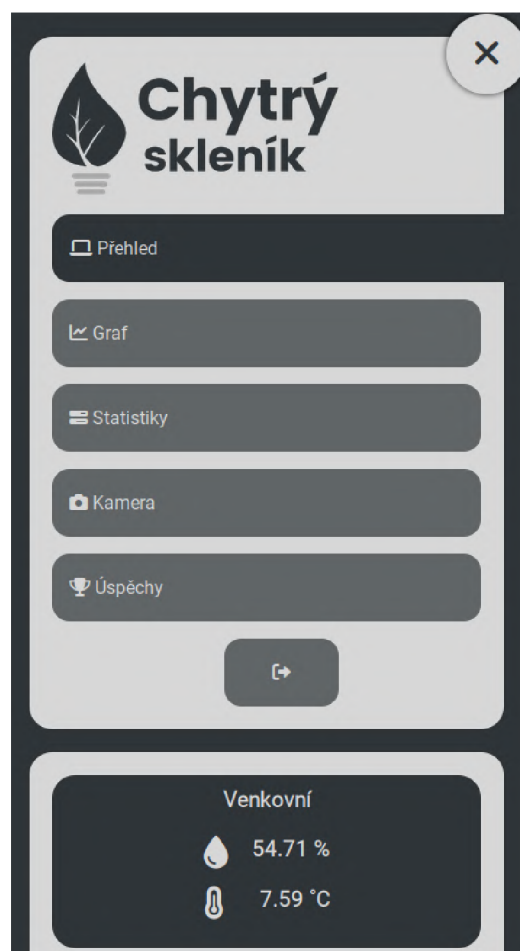
Pro webovou aplikaci byla vytvořena také verze pro mobilní zařízení. Změna zobrazení z klasické desktopové verze je nastavena na prahu 768 px. Při nižších rozlišeních již dojde ke změnám v rozložení několika elementů aplikace.

Hlavní menu není v mobilní verzi statické, ale je schováno do rozbalovací nabídky pod tlačítkem v pravé horní části obrazovky. Z přehledové sekce byla odstraněna vizualizace skleníku, který by na zařízeních s menším rozlišením nebyl dostatečně čitelný. Veškeré elementy s měřenými hodnotami jsou v přehledu seřazeny v jednom sloupci. Stejně jako v desktopové verzi se uživatel do nabídek nastavení dostane po kliknutí na tyto elementy, které jsou ovšem v mobilní verzi větších rozměrů, aby byly snadněji dostupné na dotykových obrazovkách. Řízení podmínek v postranní nabídce zůstalo stejné jako v desktopové verzi. Nabídka i s ovládacími prvky již byla navržena tak, aby vyhovovala i mobilním zařízením.

Zbylé části aplikace se dočkaly jen drobnějších úprav v rozložení prvků. Legenda ke grafu byla přeřazena do dvou sloupců namísto původních čtyř. Stejná úprava potkala i elementy ve statistikách a úspěších.



Obrázek 4.10: Úsek přehledové sekce v mobilní verzi aplikace.



Obrázek 4.11: Rozbalovací menu v mobilní verzi aplikace.

Kapitola 5

Závěr

Hlavním cílem bakalářské práce bylo navrhnout a implementovat intuitivní uživatelské rozhraní webové aplikace pro ovládání podmínek v chytrém skleníku. Aplikace je navíc doplněna o gamifikační prvky v podobě úspěchů, které se dají získat napříč celou aplikací. Zaměřují se zejména na motivaci uživatele do využívání všech funkcí, které aplikace nabízí. Díky systému úspěchů se nový uživatel orientuje v aplikaci rychleji.

Po průzkumu výhod skleníků, analýze řízení podmínek a rešerši existujících řešení byl vytvořen návrh webové aplikace. Správný návrh uživatelského rozhraní byl pro vývoj klíčový. Důraz byl kladen především na přehlednost a intuitivní ovládání. K realizaci návrhu byla využita aplikace Figma, jež dodala maketě částečnou interakci. Byl zvolen plnohodnotný návrh s barvami a interaktivními prvky, a to především pro větší přínos z uživatelského testování, které nebylo určené pouze pro odborníky ve vývoji aplikací.

Návrh následně prošel podrobným uživatelským testováním. Důležité bylo pochopit přemýšlení uživatele při hledání informací v základním menu, což zajistila metoda přiřazování kartiček. Dále proběhl řízený rozhovor nad částečně interaktivním prototypem, a nakonec uživatel hodnotil jednotlivé komponenty návrhu. Po testování proběhla optimalizace na základě zpětné vazby od respondentů.

Zadání práce bylo splněno za pomoci frontend knihovny React a backend frameworku Django. Aplikace přinesla moderní řešení pro komplexní systém řízení podmínek v chytrém skleníku. Uživatel má v přehledové sekci na očích všechny měřené údaje teplot, vlhkosti a aktuátorů. Podařilo se vytvořit minimalistický design, jenž nepřekládá uživateli příliš zahlcujících informací. Veškeré sekce s nastaveními jsou totiž skryté v postranních nabídkách. V aplikaci je možné zobrazovat naměřená data v jednom přehledném grafu. Pro přehled spotřeby elektrické energie a vody slouží statistická sekce. V grafu i statistikách je možné selektovat data pomocí kalendáře. Snímky z kamery lze volně procházet a fotografie stahovat. Do práce byly úspěšně zapracovány strukturální gamifikační prvky. Úspěchy motivují uživatele k vyzkoušení nepoznaných částí chytrého skleníku.

Webovou aplikaci by v budoucnu mohl doplnit virtuální zápisník, kam by si uživatel přidával poznámky o plodinách. Řešení by šlo obohatit o rozhraní pro detektor nemocí rostlin. Aplikaci je nutné propojit s reálným databázovým systémem, aktuálně pracuje jen s vlastními daty. Do výsledného řešení je možné dodat další ovládací prvky, kterými může být například emitor oxidu uhličitého či další zavlažovací okruhy. Systém gamifikace je nutné průběžně doplňovat o nové úkoly, aby nevymizel motivační prvek. Zároveň lze pro větší zapojení přidat i globální žebříček, který by sdružoval všechny uživatele.

Literatura

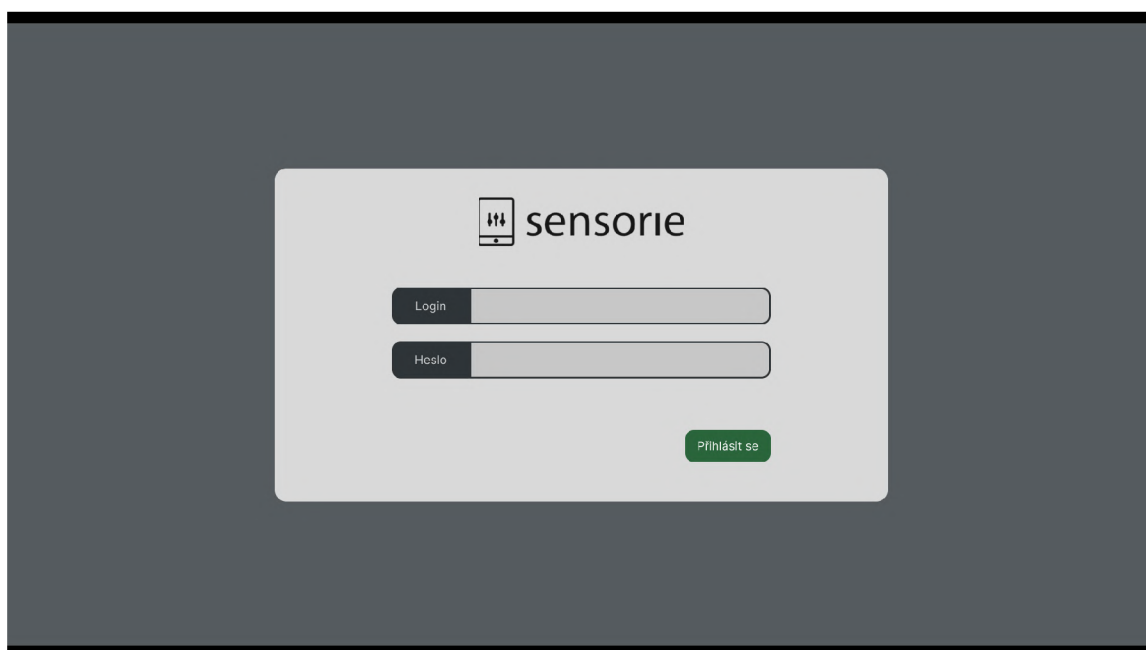
- [1] BANKS, A. a PORCELLO, E. *Learning react : functional web development with react and redux*. First edition. Sebastopol, CA: O'Reilly Media, Inc., 2017. 59–62, 81–91 s. ISBN 978-1-4919-5462-1.
- [2] BURGET, R. *Značkovací jazyky: HTML*. Vysoké učení technické v Brně, Fakulta informačních technologií, 2015 [cit. 2023-01-17].
- [3] BURGET, R. *Architektury informačních systémů*. Vysoké učení technické v Brně, Fakulta informačních technologií, 2022 [cit. 2023-04-13].
- [4] BURKE, B. *Gamify : how gamification motivates people to do extraordinary things*. Brookline, MA: Bibliomotion, 2014. 11–29 s. ISBN 978-1-937134-85-3.
- [5] BURNETT, S., BAYER, A. a IERSEL, M. V. *Ways to Use Moisture Sensors to Automate Greenhouse Irrigation* [online]. Greenhouse Grower, březen 2020 [cit. 2023-01-08]. Dostupné z: <https://www.greenhousegrower.com/technology/how-to-use-moisture-sensors-to-automate-greenhouse-irrigation/>.
- [6] BUTLER, J. *How to Heat Your Greenhouse* [online]. Gardener's Path, září 2022 [cit. 2023-01-08]. Dostupné z: <https://gardenerspath.com/how-to/greenhouses-and-coldframes/greenhouse-heating/>.
- [7] CAMPOTTI, C. A., LATINI, A., MATTEO, S. a VIOLA, C. *Certificati Bianchi: Guida Operativa per la Scheda 40E—Gli Incentivi del DM 28 Dicembre 2012 sui Sistemi Serra*. Rome, Italy: ENEA, 2014.
- [8] DEBENHAM, L. *Growing Crops all Year Round in Your Greenhouse* [online]. Greenhouse Growing, červenec 2021 [cit. 2023-01-08]. Dostupné z: <https://www.greenhousegrowing.co.uk/growing-vegetables/growing-crops-year-round-your-greenhouse/>.
- [9] FLANAGAN, D. *JavaScript: the definitive guide*. Fifth edition. Sebastopol, CA: O'Reilly Media, Inc., 2006. ISBN 978-0-59-610199-2.
- [10] GREENHOUSE EMPORIUM. *Greenhouse Irrigation – What's the best watering system?* [online]. Greenhouse Emporium, duben 2021 [cit. 2023-01-08]. Dostupné z: <https://greenhouseemporium.com/blogs/greenhouse-gardening/greenhouse-irrigation-systems/>.
- [11] HU, G. a YOU, F. Renewable energy-powered semi-closed greenhouse for sustainable crop production using model predictive control and machine learning for energy management. *Renewable and Sustainable Energy Reviews*. Elsevier. Říjen 2022, sv. 168, s. 458–468. DOI: 10.1016/j.rser.2022.112790.

- [12] IONOS. *What is CSS? Definition and application* [online]. IONOS Digital Guide, červenec 2021 [cit. 2023-01-17]. Dostupné z: <https://www.ionos.com/digitalguide/websites/web-design/what-is-css/>.
- [13] ISABEL. *Heat mats for plants: What you need to know* [online]. Greenhouse Hunt, srpen 2020 [cit. 2022-01-08]. Dostupné z: <https://www.greenhousehunt.com/greenhouse-supplies/heat-mats-for-plants/>.
- [14] KAPP, K. M., BLAIR, L. a MESCH, R. *The gamification of learning and instruction fieldbook : ideas into practice*. First edition. San Francisco: Wiley, 2014. 54–66 s. ISBN 978-1-118-67443-7.
- [15] KRUG, S. *Don't make me think, revisited : a common sense approach to web usability*. Third edition. San Francisco: New Riders, 2014. ISBN 978-0321965516.
- [16] LAMBERS, H. a BASSHAM, J. A. *Photosynthesis* [online]. Encyclopedia Britannica, n.d. [cit. 2023-01-10]. Dostupné z: <https://www.britannica.com/science/photosynthesis>.
- [17] LEE, A. *Pros & Cons of Django Framework* [online]. Code Condo, říjen 2020 [cit. 2023-01-16]. Dostupné z: <https://codecondo.com/pros-cons-of-django-framework/>.
- [18] LI, G., TANG, L., ZHANG, X., DONG, J. a XIAO, M. Factors affecting greenhouse microclimate and its regulating techniques: A review. *IOP Conference Series: Earth and Environmental Science*. IOP Publishing. Červen 2018, sv. 167, č. 1. DOI: 10.1088/1755-1315/167/1/012019.
- [19] MACHINATIONS. *What is game economy design* [online]. Machinations, listopad 2021 [cit. 2023-01-18]. Dostupné z: <https://machinations.io/articles/what-is-game-economy-design/>.
- [20] MONK, J. *The Pros and Cons of Greenhouses* [online]. Greenhouse Info, prosinec 2020 [cit. 2023-01-08]. Dostupné z: <https://greenhouseinfo.com/pros-cons-greenhouses/>.
- [21] NSW GOVERNMENT. *Light in greenhouses* [online]. NSW Department of Industry, červen 2007 [cit. 2023-04-13]. Dostupné z: <https://www.dpi.nsw.gov.au/agriculture/horticulture/greenhouse/structures-and-technology/light>.
- [22] QUOKKA LABS. *TypeScript vs. JavaScript: Which One Should Use, and Why?* [online]. Medium, prosinec 2022 [cit. 2023-04-12]. Dostupné z: <https://medium.com/tech-and-tricks/typescript-vs-javascript-which-one-should-use-and-why-c0c766454dfe>.
- [23] RANJAN, R. *What is a Framework in Programming & Why You Should Use One* [online]. Net Solutions, říjen 2021 [cit. 2023-01-16]. Dostupné z: <https://www.netsolutions.com/insights/what-is-a-framework-in-programming/>.
- [24] SETHI, V. On the selection of shape and orientation of a greenhouse: Thermal modeling and experimental validation. *Solar Energy*. 2009, sv. 83, č. 1, s. 21–38. DOI: <https://doi.org/10.1016/j.solener.2008.05.018>. ISSN 0038-092X.

- [25] SIDDHARTH. *Angular: Pros and Cons* [online]. DEV Community, srpen 2021 [cit. 2023-04-12]. Dostupné z: <https://dev.to/siddharthshyniben/angular-pros-and-cons-m91>.
- [26] SLASH_IT. *SASS / Introduction* [online]. GeekForGeeks, leden 2022 [cit. 2023-01-17]. Dostupné z: <https://www.geeksforgeeks.org/sass-introduction/>.
- [27] STACKOVERFLOW. *Stack Overflow Developer Survey 2021* [online]. Stackoverflow, říjen 2021 [cit. 2023-01-16]. Dostupné z: <https://insights.stackoverflow.com/survey/2021#section-most-popular-technologies-web-frameworks>.

Příloha A

Návrh uživatelského rozhraní



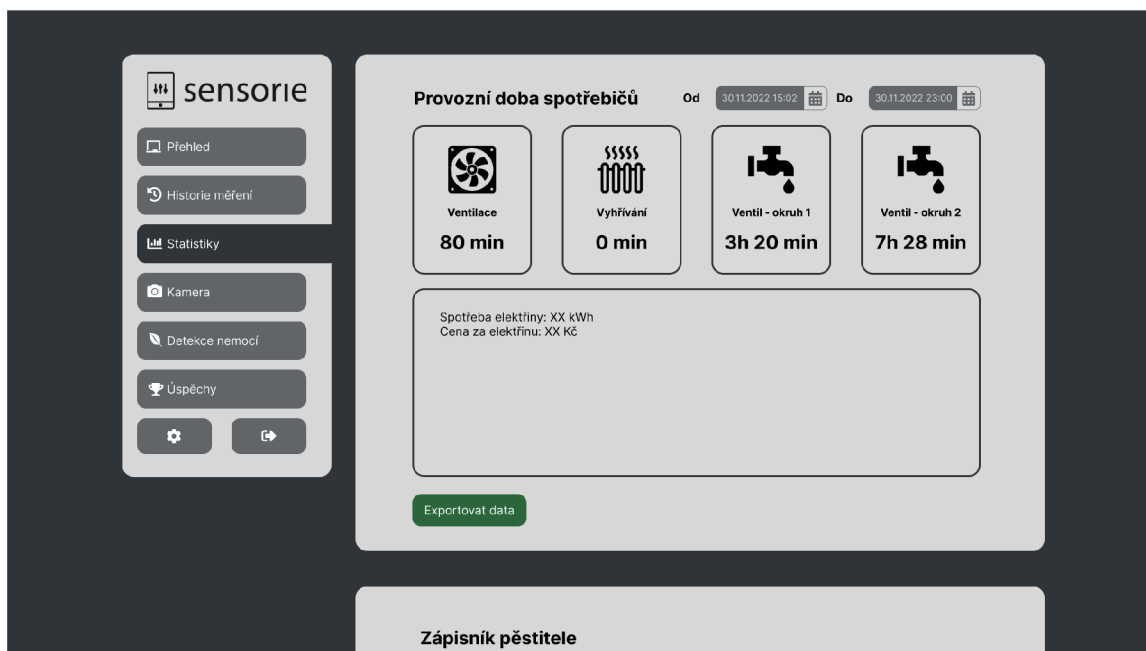
Obrázek A.1: Návrh přihlašovací obrazovky.



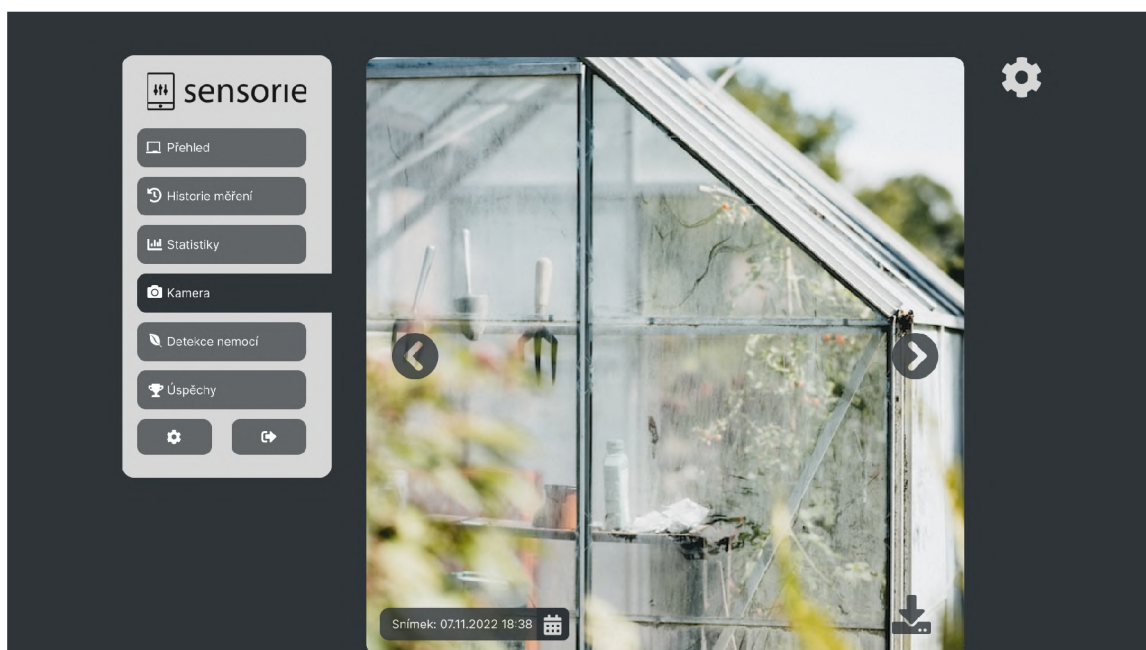
Obrázek A.2: Návrh přehledové sekce.



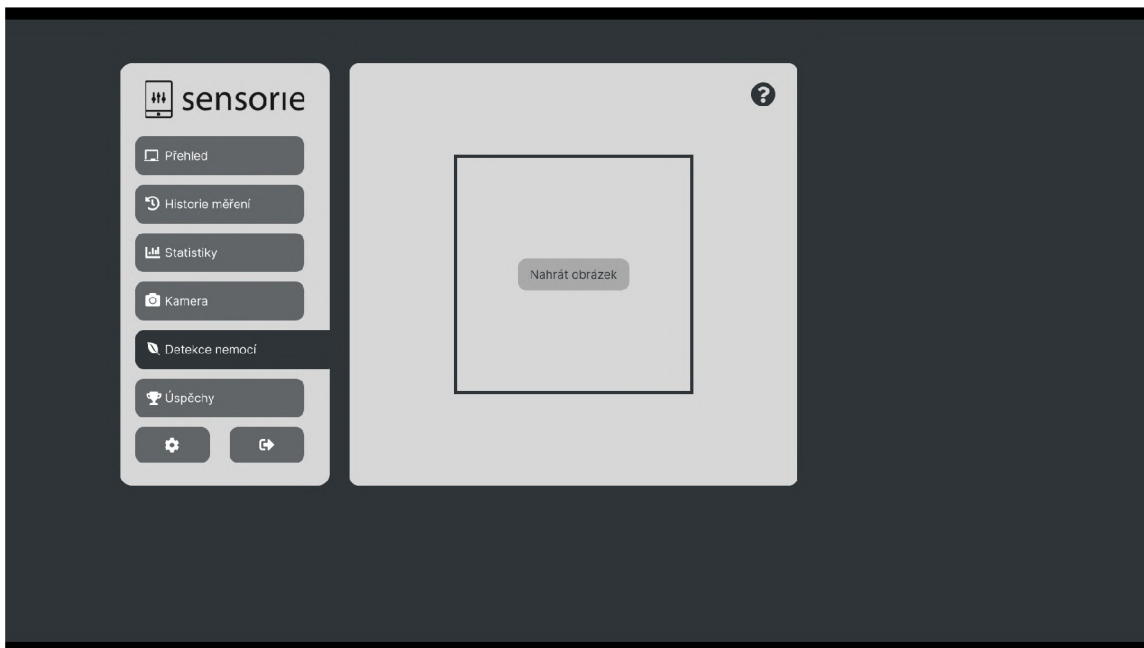
Obrázek A.3: Návrh sekce Historie měření. Zobrazení třech hodnot v grafu.



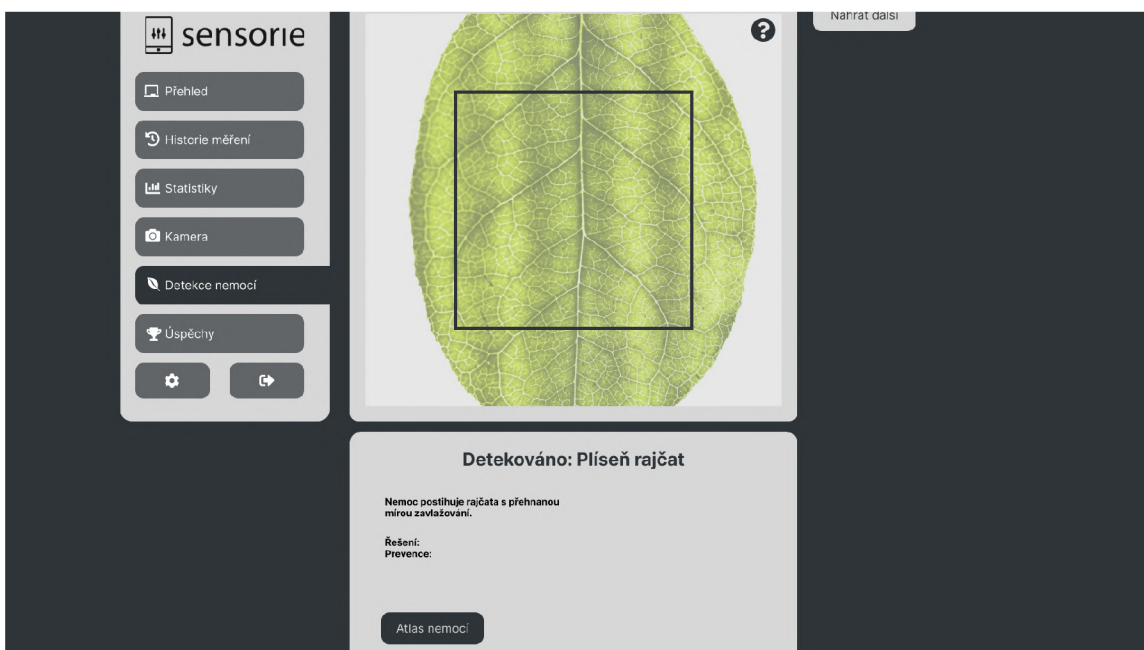
Obrázek A.4: Návrh statistické sekce.



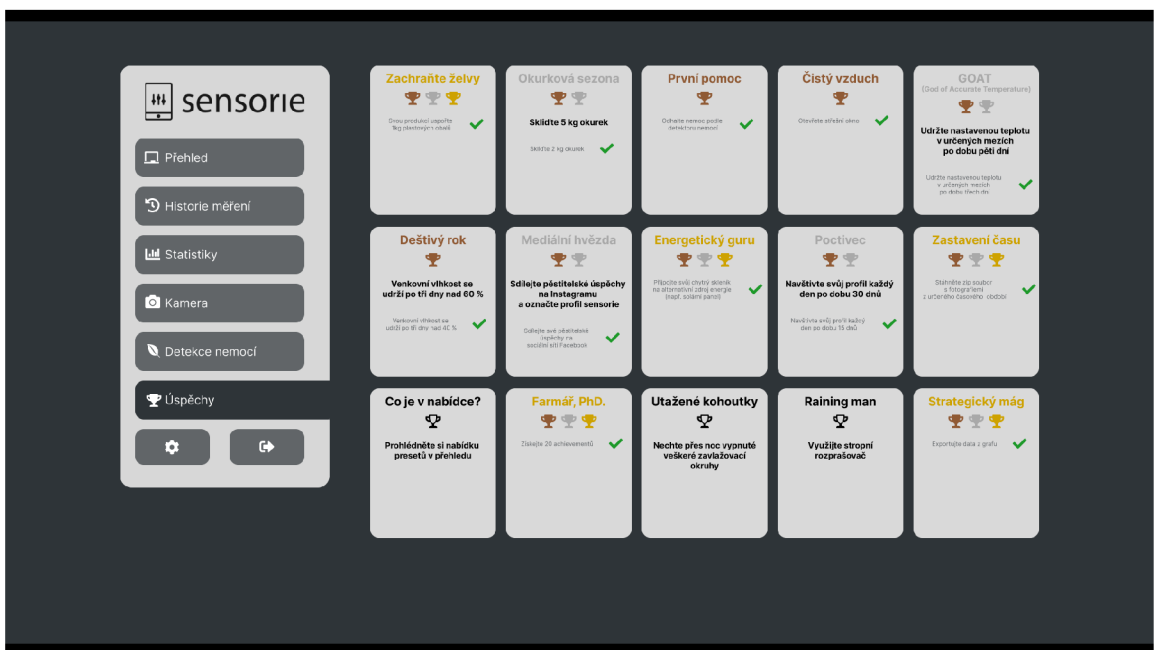
Obrázek A.5: Návrh zobrazení snímků z dohledové kamery.



Obrázek A.6: Návrh uživatelského rozhraní pro výstup z detektoru nemocí. Stav před nahráním fotografie.



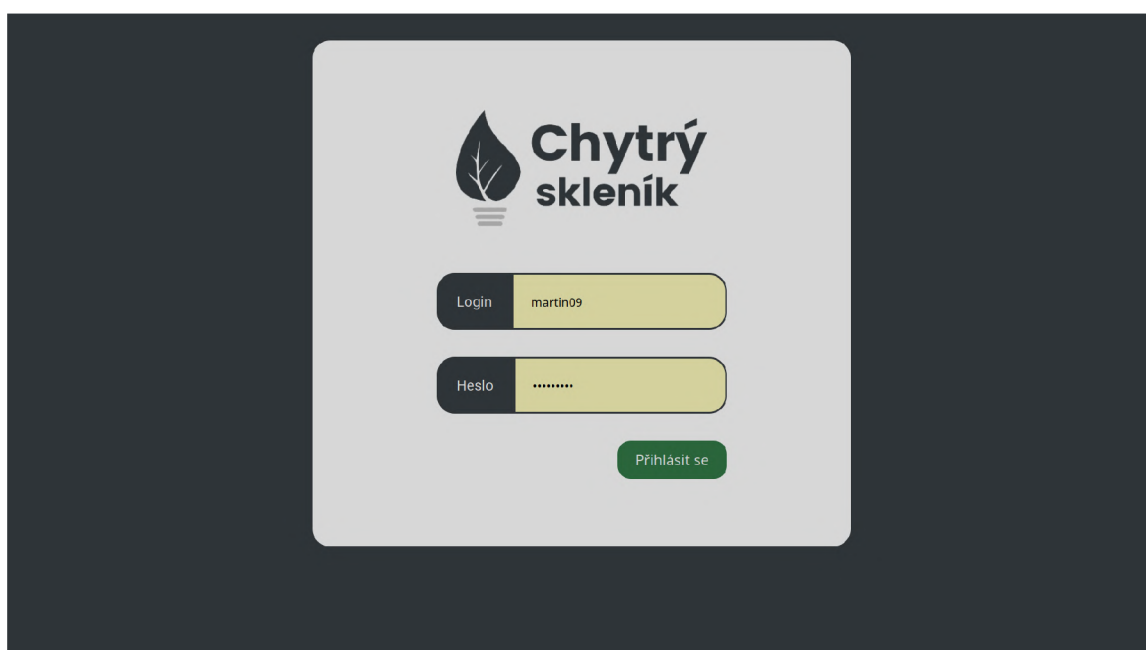
Obrázek A.7: Návrh uživatelského rozhraní pro výstup z detektoru nemocí. Stav po nahrání fotografie a dokončení detekce.



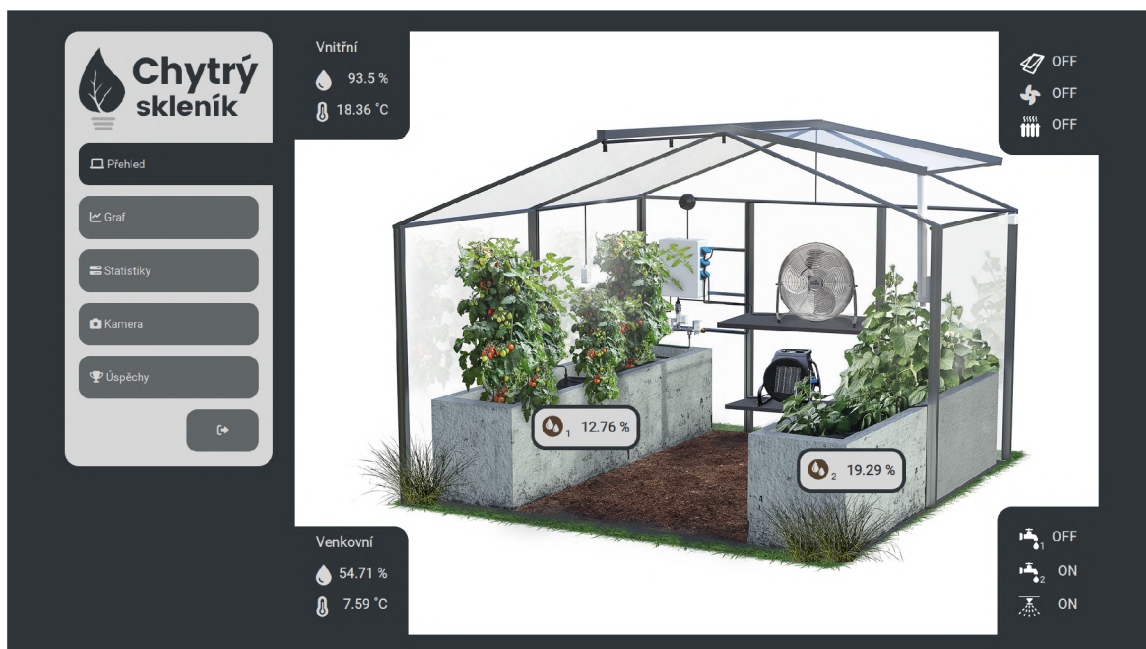
Obrázek A.8: Návrh sekce pro zobrazení všech úspěchů. Ukázka úspěchů v různých stavech.

Příloha B

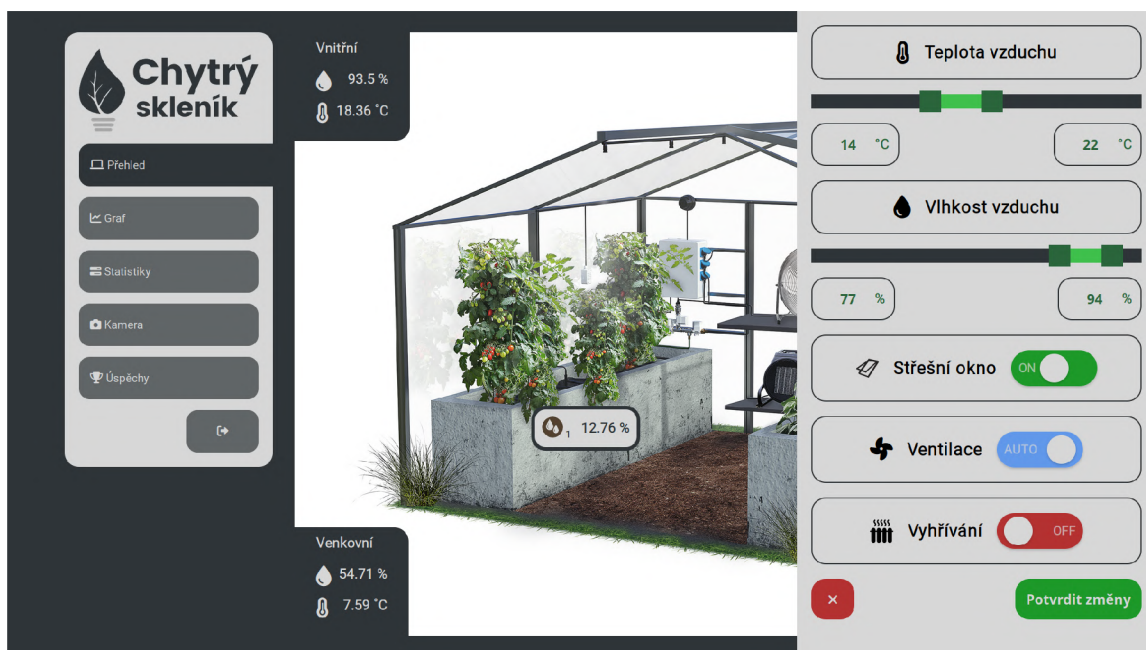
Snímky aplikace



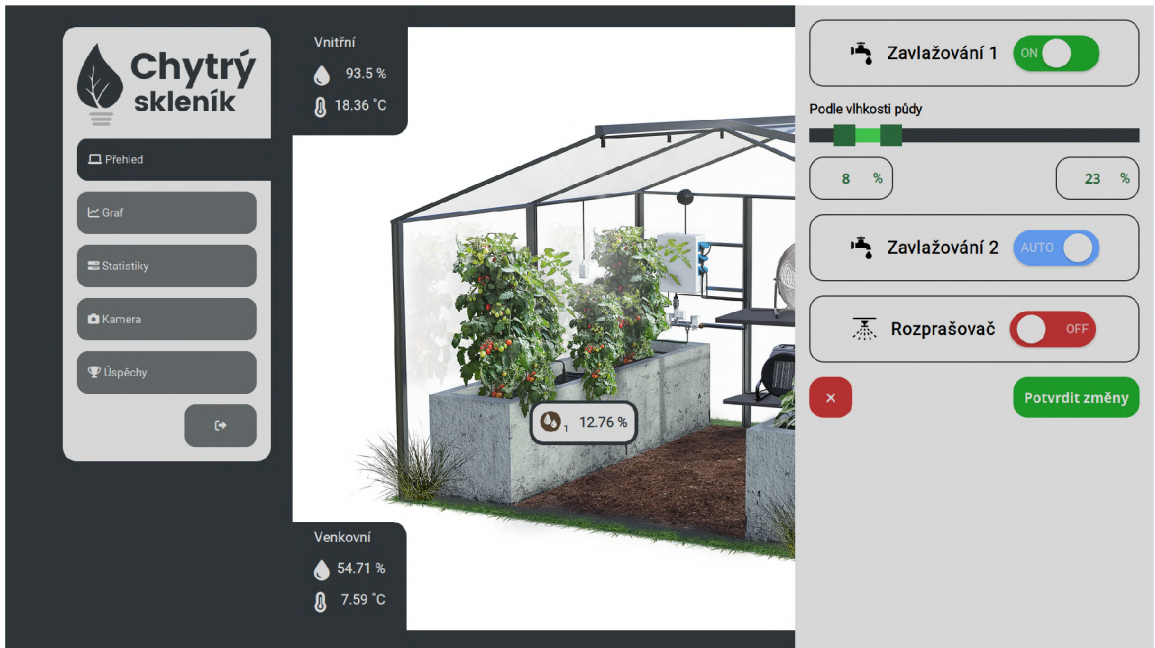
Obrázek B.1: **Přihlašovací obrazovka.** Automaticky předvyplněné přihlašovací údaje.



Obrázek B.2: Úvodní přehledová obrazovka.



Obrázek B.3: Úvodní přehledová obrazovka s nastavením zařízení.



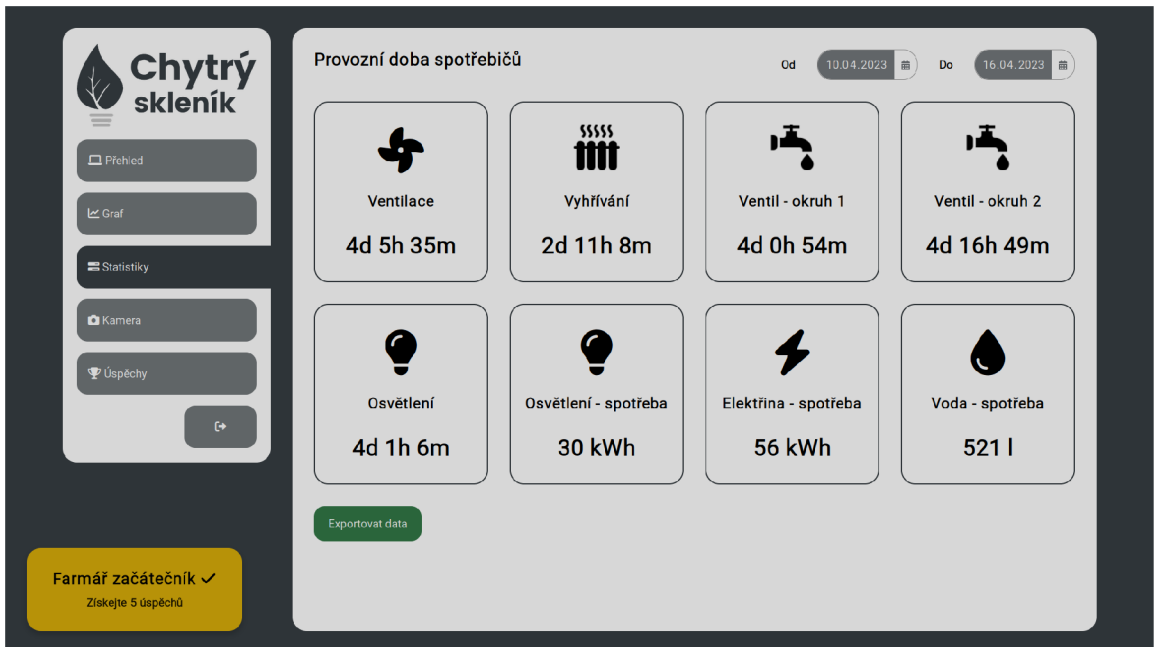
Obrázek B.4: Úvodní přehledová obrazovka s nastavením zavlažování.



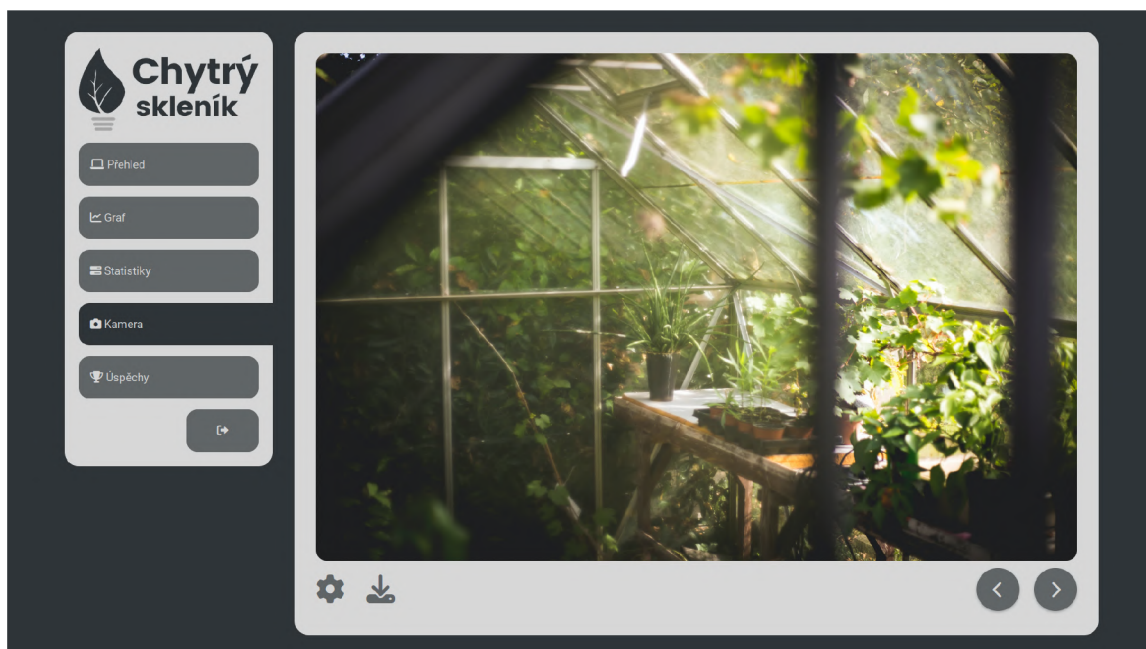
Obrázek B.5: **Graf.** Zobrazení čtyř hodnot v jednom grafu.



Obrázek B.6: **Graf.** Ukázka analýzy změn zavlažování na vlhkosti vzduchu i půdy.



Obrázek B.7: **Statistiky.** Výběr naměřených dat za poslední týden. V levé dolní části ukázka notifikace o splněném úkolu.



Obrázek B.8: **Kamera.** Výběr poslední pořízené fotografie z dohledové kamery.



Obrázek B.9: **Úspěchy.** Ukázka více úkolů ze sekce Úspěchy.