

KATTEGATTGYMNASIET

Gymnasiearbete  
Theodor Lundqvist  
Läsåret 18-19

# **Programmering**

## **Apputveckling**

Handledare: Stefan Fredriksson

## **ABSTRACT**

This paper describes the process of developing a mobile application from scratch, the methods chosen and how they were implemented. The primary task was to build a functional mobile application with some predefined functions. The purpose of the app was to provide lunch menus in an easy and accessible manner with one of the key points being to provide them sorted by dishes instead of by restaurants as some other apps and web pages do.

The paper also describes the development of the supporting functions needed by the app such as cloud storage and a web page for editing restaurant information.

**Keywords:** *App development, webb development, IOS, Android, Database, Software, Server, cloud storage*

## **INNEHÅLLSFÖRTECKNING**

<b>1. INLEDNING</b>	4
1.1 BAKGRUND	4
1.2 SYFTE	4
1.3 FRÅGESTÄLLNINGAR	4
1.4 AVGRÄNSNINGAR	4
<b>2. METOD</b>	5
2.1 PLANERING	5
2.2 UTFÖRANDE	5
2.2.1 PROGRAMVARA	14
3.1 SVAR PÅ FRÅGESTÄLLNINGAR	15
<b>4. DISKUSSION OCH ANALYS</b>	16
4.1 FÖRBÄTTRINGAR	16
4.2 INKÖP	16
4.3 PROGRAMVARA	17
4.4 RAPPORTARBETE	17
<b>5.KÄLL- OCH LITTERATURFÖRTECKNING</b>	18
5.1 WEBBKÄLLOR	18
5.2 VIDEOKÄLLOR	19
<b>6. BILAGOR</b>	20
6.1 APP ANDROID UTVECKLING	20
6.2 APP IOS	23
6.3 HEMSIDA	24
6.4 VIDEOR HEMSIDA	25
6.5 LOGGBOK	25

# 1. INLEDNING

## 1.1 BAKGRUND

Jag är medlem i ett UF företag som heter Lunchdax UF. Vi var ofta ute och åt på stan, men hade alltid svårt att bestämma oss för var vi skulle äta. Vi bestämde oss för att vi ville göra en app där man kan se alla närliggande lunchmenyer på ett och samma ställe. Eftersom jag var den av oss som hade mest erfarenhet inom programmering blev det jag som fick utveckla appen. Då det var ett stort projekt valde jag att även använda uppgiften till mitt gymnasiearbete. Då jag endast hade erfarenhet av att göra mindre spel skulle detta bli en stor utmaning och jag har fått lära mig mycket nytt på vägen. Jag valde även detta ämnet då jag tyckte det var intressant att programmera och ville lära mig mer.

## 1.2 SYFTE

Syftet är att lära mig mer om programmering genom att utveckla och lansera en fungerande app i samarbete med UF-företaget Lunchdax UF. Detta ger en djupare förståelse om ämnet programmering samt visar vilka kunskaper jag bär med mig från min utbildning på Kattegatt.

## 1.3 FRÅGESTÄLLNINGAR

*Hur utvecklar man en app?*

*Hur ska datan lagras?*

*Hur kan restaurangerna själva redigera sina rätter?*

## 1.4 AVGRÄNSNINGAR

Arbetet avgränsas till följande delar för att kunna genomföras i tid för inlämning.

- En app till Android och IOS som kan visa inlagda rätter samt avstånd och vägbeskrivning.
- Ett system som möjliggör för oss eller restauranger att lägga till och redigera rätter.

Vad jag väljer att inte inkludera är vissa funktioner i appen såsom sökfunktioner och att kunna få mer information om olika rätter och restauranger.

## 2. METOD

### 2.1 PLANERING

Jag visste när jag började att detta skulle bli ett stort projekt, men jag hade ingen aning om vad som egentligen skulle komma att krävas då jag hade väldigt lite erfarenhet. Jag lade dock upp en plan för hur jag tänkte mig att systemet skulle fungera. Jag visste att vi skulle behöva någon typ av server att lagra data om restaurangerna på samt en app som hade åtkomst till detta. I planen ingick även att appen lokalt skulle söka upp närliggande restauranger via Google Maps och sedan fråga servern om det finns menyer till dessa restauranger tillgängliga.

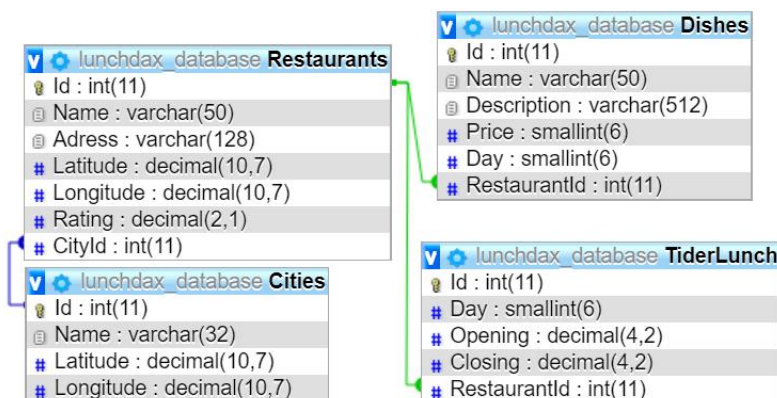
Då jag visste att jag hade mycket att göra satte jag genast igång att bryta ner problem efter problem.

### 2.2 UTFÖRANDE

Det första jag gjorde var att undersöka huruvida det gick att utveckla en app som fungerar både på Android och IOS så jag slapp att bygga två olika. Vad jag kom fram till var att jag kunde använda mig av Xamarin för att lyckas med detta. Tyvärr visade sig detta inte riktigt stämma vilket jag märkte efter många timmars experiment och informationssökande. Jag fick nämligen felkod efter felkod när jag försökte installera de paket som krävdes till android. IOS fungerade inte heller då jag behövde en Apple dator för att ens testa den. Det visade sig att en Apple dator skulle komma att krävas oavsett utvecklingssätt vilket ledde till att jag påbörjade söka efter en sådan.

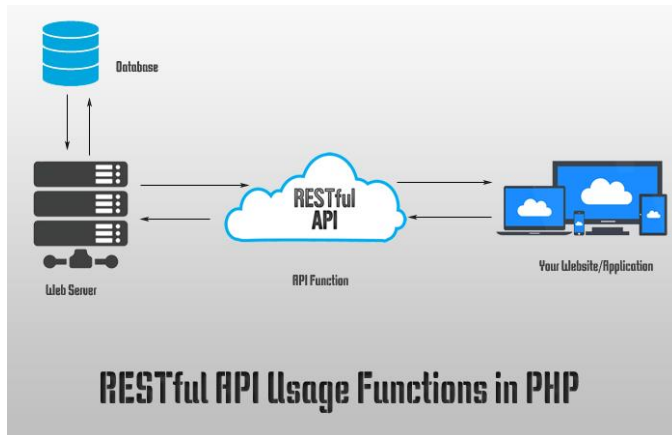
Efter att ha gett upp på cross-platform Xamarin så börjar jag att undersöka hur man bygger en databas där jag kan lagra mina restauranger och rätter. Jag går igenom en tutorial på youtube söker runt på nätet och har ett möte med Stefan Fredriksson, programmeringslärare.

Efter detta har jag fått ihop en databasdesign som jag är nöjd med. (Bilaga 1)



Bilaga 1 - databasstruktur.

När jag hade fått ihop en databas blev jag återigen osäker på vad jag skulle göra och bokar ett möte med Nils-olof Bankell som är CTO för Jayway i Halmstad. Han ger en mycket bra grundläggande förståelse för hur webbkommunikation mellan databaser och appar går till och ger mig tips på att använda php eller ruby för att bygga den backend som ska sköta kommunikationen mellan app och databas beskrivet i *bilaga 2*.



*Bilaga 2 - serverstruktur.*

Jag undersöker dessa språken och bestämmer mig för att använda PHP. Jag bokar in ett möte med skolans lärare inom webbutveckling, Fredric Persson för att få reda på mer om detta språket. Under tiden undersöker jag Googles API för platser. Efter mötet har jag fått en bra grundförståelse för php och börjar bygga en backend. Jag fortsätter att testa google maps API och får mitt program att spotta ut alla restauranger i närheten av min position. Jag inser också att det kostar pengar för varje fråga man skickar till google. Jag behöver ta reda på exakt vilka utgifter jag kan få av detta. Det visar sig att det är rätt dyrt att använda sig av deras API. Då min första tanke var att appen skulle söka efter restauranger i närheten med hjälp av Googles API och sedan kolla vilka restauranger som matchar med de i vår databas. Hade appen 1000 användare per dag hade detta kostat minst 9000 kr/månad vilket var alldeles för mycket vilket ledde till att jag fick tänka om.

Jag utvecklar en backend som spottar ut alla rätter ur databasen som ligger på en restaurang nära de koordinater man ger den. När backenden fungerar börjar jag utveckla appen till Android för att kunna visa de rätter som hämtas från servern. Det första jag gör är att följa några tutorials på youtube för att förstå hur android refererar till knappar och bilder som visas på appen. Jag designar därefter hur appen ska se ut och får den att ladda ner rätterna från servern samt visa dem i en lista. Jag fortsätter med att förfinat utseendet på appen samt lägga till funktioner såsom att uppdatera, visa avstånd och få vägbeskrivning.

Jag börjar utveckla en hemsida där jag ska kunna redigera restaurangerna som ligger i vårt system på ett enkelt sätt. (*Bilaga 3*) Det är väldigt omständligt och tar mycket tid. Slutresultat blev enligt bilaga 3, inte så snyggt eller snabbt men det skulle fungera för oss. Designen gjorde Gabriel Bexelius medans jag gjorde programmet som skulle hämta rätt restauranger och föra in rätt värde i rätt tabell. Nu började jag få problem med hastigheten på servrarna, ibland tog det över 10 sekunder att ladda in hemsidan vilket man var tvungen att göra varje gång något värde skulle uppdateras då jag skrev programmet i PHP. Jag började också få problem med säkerheten i detta programmet då jag även ville göra en version där

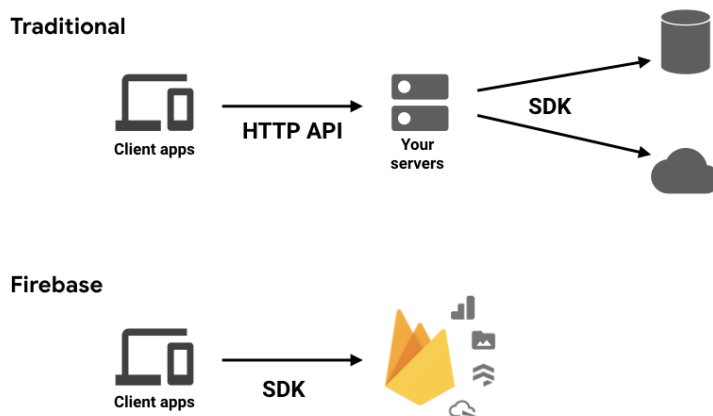
restaurangerna själva kunde redigera sina menyer och inte hade någon erfarenhet inom detta. Jag såg brister i säkerheten överallt och det tog längre och längre tid att täta alla hål.

The image shows three separate web forms for editing restaurant data. Each form has a title, a dropdown menu, and a 'Hämta' button. Below each form is a larger form with multiple input fields and a 'Spara' button.

- Stad**: A dropdown menu with a 'Hämta' button. Below it, a form with fields for 'Namn:', 'Lat:', 'Long:', and a 'Spara' button.
- Restaurang**: A dropdown menu with a 'Hämta' button. Below it, a form with fields for 'Namn:', 'Sök', 'Adress:', 'Lat:', 'Long:', 'Rating:', and a 'Spara' button.
- Maträtt**: A dropdown menu with a 'Hämta' button. Below it, a form with fields for 'Namn:', 'Beskrivning:', 'Pris:', 'Dag:', 'Måndag' (dropdown), and a 'Spara' button.

*Bilaga 3 - hemsida för redigering av restauranger.*

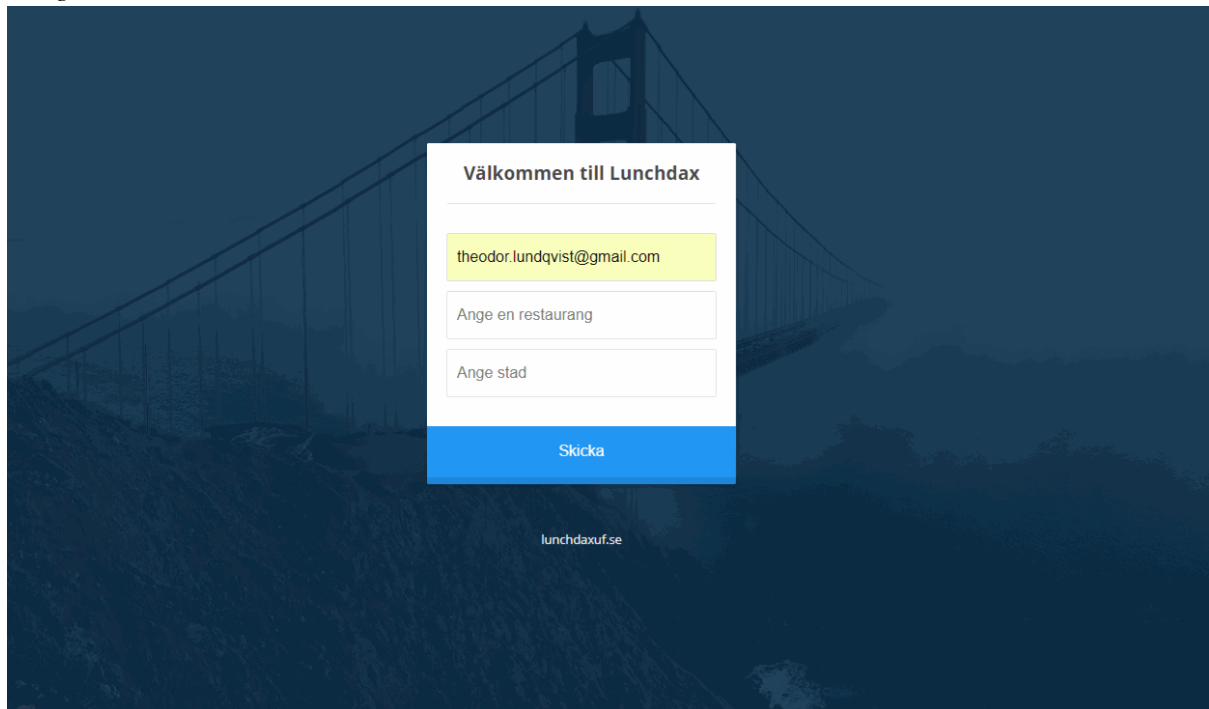
Efter allt strul som jag hade med detta började jag fundera på andra lösningar. I första hand kollade jag efter andra serverlösningar då skolans server var så seg. Jag ville också testa javascript då detta verkade mycket smidigare för frontend applikationer än PHP. Jag var dock rädd för att säkerheten i javascript skulle vara dålig då användaren själv kan ändra koden och skicka vilka värden som helst till servern. Jag hittade Google Firebase som utlovade snabb och säker backend utan underhåll samt ytterligare funktioner såsom säker inloggning via facebook, google, github, telefonnummer, mail mm. (Bilaga 4)



*Bilaga 4 - Standard serverstruktur vs. Firebase.*

Detta skulle dock innebära att jag var tvungen att börja om från början utan att ha användning för de 50 timmar jag hade lagt på databas/backend utveckling. Jag hade dock lärt mig mycket och såg många fördelar med att använda Firebase. Data i Firebase sparas i ett träd av dokument istället för i olika "excel ark" som i MySQL. Efter att ha kommit från MySQL var detta lite svårt att ta in men var egentligen mer naturligt för den data struktur som jag hade. En stad innehåller restauranger som innehåller maträtter. Vi hade en ny ide på hur menyer skulle redigeras så under tiden Gabriel slutförde designen på den nya hemsidan skapade jag ett inloggningssystem. Vi som admin hade nu tillgång till

en sida där vi kan skapa nya restauranger. Vi skriver namnet på restaurangen och mailen som kunna redigera den restaurangen. Via Google hittar sidan automatiskt restaurangen och lägger in allt från position till öppettider automatiskt. Ett mail skickas sedan ut till den angivna mailen med en länk där de kan skapa ett lösenord, därefter loggar de direkt in och kan börja redigera sina rätter. *Bilaga 5, Bilaga 6.*



*Bilaga 5 - skapar ett konto och loggar in på detta.*

#### Ditt konto på Lunchdax



Lunchdax <automail@lunchdaxuf.se>  
2019-01-25 20:54



Till: theodor.lundqvist@gmail.com



Hej!

Vi är nu stolta att presentera appen **Lunchdax** med tillhörande hemsida där du som administratör snabbt och enkelt kan redigera dina menyer.

Vi har skapat ett konto åt dig med följande mail: [theodor.lundqvist@gmail.com](mailto:theodor.lundqvist@gmail.com). När du har loggat in kan du direkt börja lägga upp dina rätter.

[Tryck här för att skapa ett lösenord till ditt konto](#)

För tillfället går det inte att logga in i IE < 9 (Versioner av Internet Explorer Från 2012 eller tidigare) Var vänlig att öppna sidan i Safari eller Google Chrome för bästa kompatibilitet.

Om du har frågor når du oss på [lunchdaxuf@gmail.com](mailto:lunchdaxuf@gmail.com)

Tack

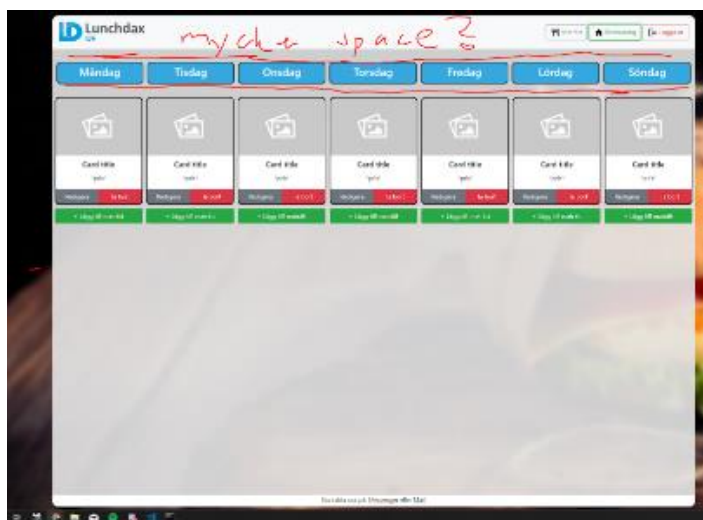
Teamet bakom **Lunchdax**

*Bilaga 6 - Detta mail tar restaurangägaren emot när ett konto skapas med deras mail.*



Eftersom att designen på webbsidan fortfarande inte var klar. (Bilaga 7) Började jag att utveckla ett verktyg som skulle användas av webbsidans användare för att ladda upp bilder på restaurangen eller rätterna. Jag ville dock inte behöva skicka bilder i hög upplösning till appen då de utseende mässigt var rätt små. Viktigt var också att bilderna var helt fyrkantiga så att det såg snyggt ut. Verktöget låter användaren dra och släppa en bild i vilken storlek som helst för att sedan klippa bilden till 1:1 och ändra upplösningen till 480x480 eller största möjliga ifall bilden är mindre. Jag lade också till en funktion som låter användaren lägga in en länk till en bild istället för att behöva ladda ner den.

Jag börjar också att utveckla appen till IOS. Jag får den att ladda ner rätterna från servern och visa dem i en lista av kort. Jag sorterar även på avstånd och gör en knapp som skickar användaren till en vägbeskrivning. Appen till IOS blev lite lättare att utveckla i och med att jag skulle göra ungefär samma sak som jag hade gjort på android bara i ett annat språk. Detta gjorde också att appen till IOS blev lite mer raffinerad då jag kunde ta bort buggar och onödigheter när jag skrev om hela koden. (Bilaga 8)



Bilaga 7 - Ofärdig design på hemsida.




*Bilaga 8 - Tidig bild på IOS app.*

Jag laddar upp Android appen till Google play vilket totalt tog cirka en timme. Jag försöker sedan ladda upp appen till IOS vilket de hade lyckats göra riktigt osmidigt. Appen ska först byggas ihop vilket tar 20 minuter. Efter det ska den laddas upp till Apples hemsida vilket tar ytterligare 20 minuter. Det är efter dessa 40 minuter man får ett felmeddelande att en av 20 bilder man behövde ha på appikonen var i fel storlek. När man väl har fått bort alla felmeddelanden ska någon i USA undersöka om appen är bra nog vilken tar 2 dagar till. Totalt nekar de appen 4 gånger med olika små fixar som behöver göras som orsak vilket gör att vi inte hinner lansera appen till skolans första ufmässa.

Vid denna tiden har designen på hemsidan blivit tillräckligt bra för att jag ska kunna börja lägga in lite funktionalitet. Jag börjar med att ladda in rätterna för den restaurangen man är inloggad på för att sedan lägga in funktioner som gör att man kan ändra redan inlagda rätter samt skapa nya. (Bilaga 9). Med mitt tidigare byggda verktyg löser jag så att man kan ladda upp bilder. (Bilaga 9).

Din restaurang



Välj en fil

eller

http://Länka.din.bild

eller

Dra och släpp filen på bilden

Namn:

Halmstad Live

Adress:

Karl XIs väg 47, 302 95 Halmstad, Sverige

Öppettider

Mån:

Öppnar

till

Stänger

och

Öppnar

till

Stänger

Tis:

10:00

till

16:00

och

16:40

till

22:30

Ons:

10:00

till

22:00

och

Öppnar

till

Stänger

Tor:

10:00

till

22:00

och

Öppnar

till

Stänger

Fre:

10:00

till

23:00

och

Öppnar

till

Stänger

Lör:

10:00

till

23:00

och

Öppnar

till

Stänger

Sön:

Öppnar

till

Stänger

och

Öppnar

till


Stänger

\* Tiderna som är blåmarkerade behöver du bara fylla i om du har flera öppettider på samma dag.

Stäng

Spara

Redigera din maträtt



Välj en fil

eller

http://Länka.din.bild

eller

Dra och släpp filen på bilden

Namn:

Buffé

Pris:

99

Beskrivning:

1. Raggmunk och fläsk (LF) 2. Het chili con carne med ris (LF, GF) 3. Vegetarisk kålpudding med gräddsås, kokt

☐ Vegetariskt

☐ Veganskt

☐ Glutenfritt

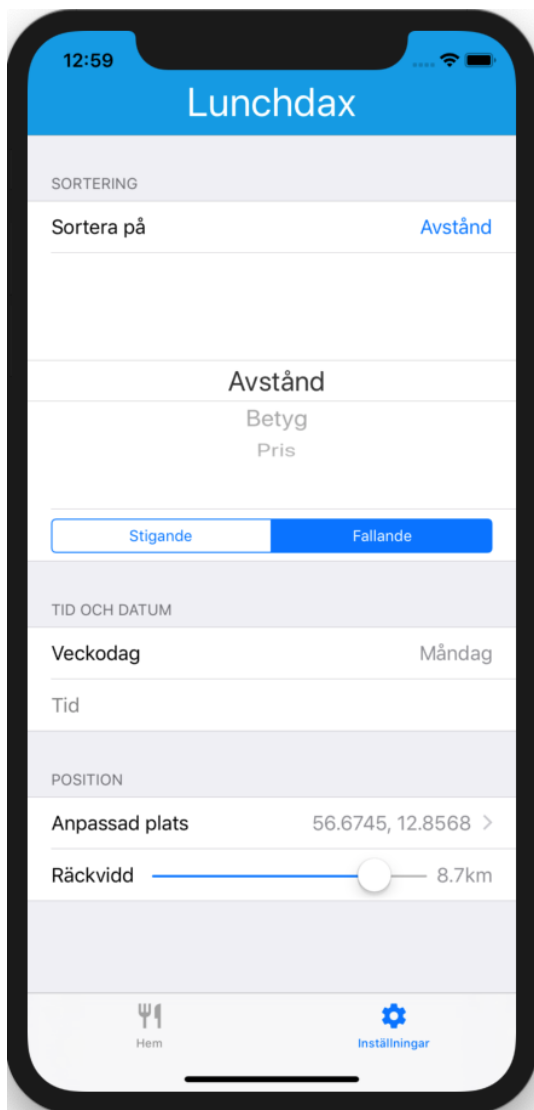
☐ Laktosfritt

Stäng

Spara

Bilaga 9 - redigera restaurang (vänster) och rätt (höger).

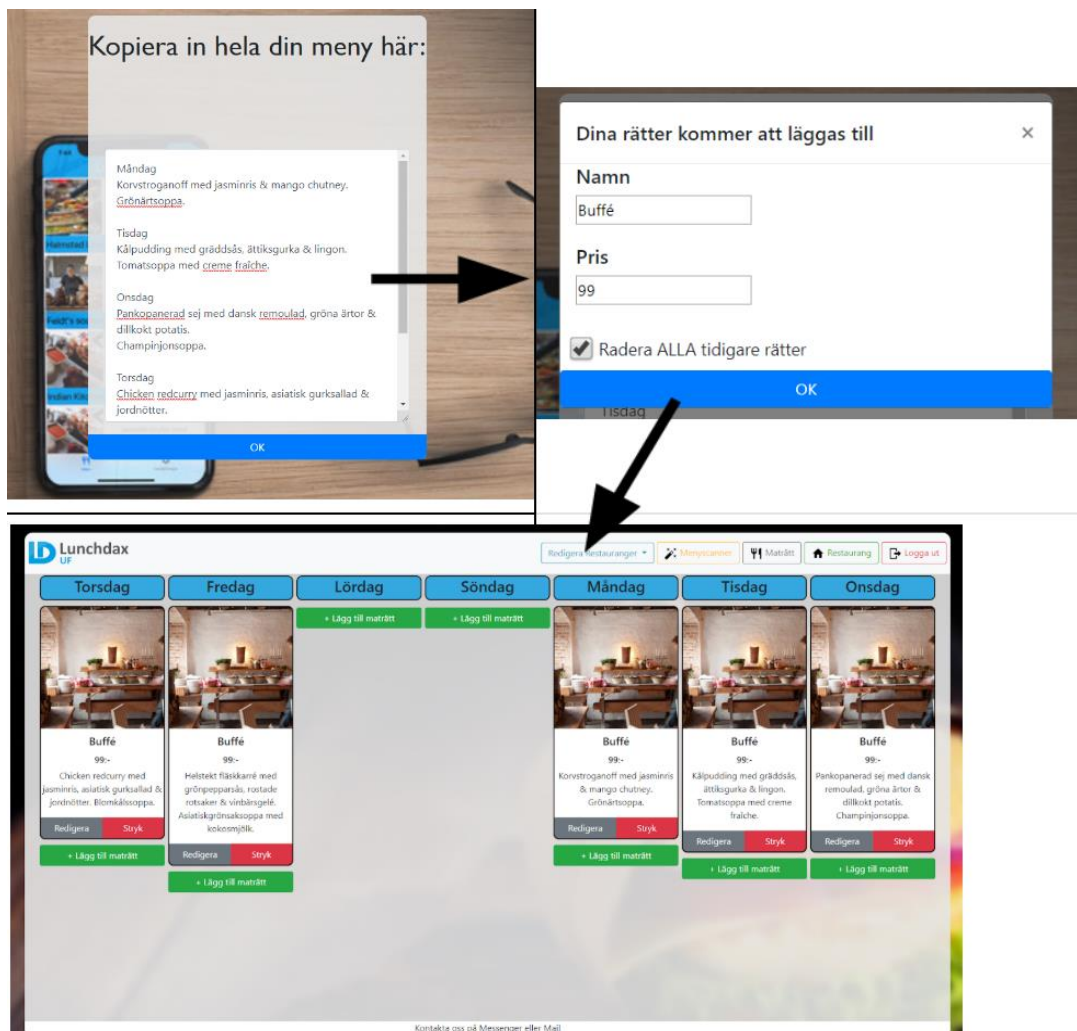
Efter många tester och småfixar är jag nöjd för tillfället och börjar istället med att fixa de klagomål som Apple hade på vår app. Jag lägger till mer funktionalitet såsom sortering på avstånd, pris och betyg som filtrering på veckodag och plats. (Bilaga 10). För att åstadkomma detta använder jag ett verktyg från github vid namn [Eureka](#). Detta gör också att Apple äntligen godkänner appen. Det tar dock flera dagar för dem att svara vilket gör att lanseringen dröjer till efter skolans första UF-mässa.



*Bilaga 10 - ny sida i IOS app för bl.a. sortering.*

Vid denna tidpunkten är jag nöjd med hela systemets funktionsduglighet och börjar pillra med mindre grejer som jag har lagt undan i förmån för viktigare delar. Jag löser så att "kontakta oss" på vår hemsida faktiskt fungerar vilket innebär att en besökare nu kan fylla i några fält, godkännas av recaptcha och skicka ett mail till oss genom vår hemsida. När jag väl pillrar med mailen ser jag till att vi som administratörer får mail varje gång någon rätt eller restaurang uppdateras för att ha koll på nya uppdateringar samt veta att inget olämpligt laddas upp. För tillfället hade dock bara restaurangen själv möjlighet att redigera sina menyer vilket gjorde att vår vetskap om potentiell uppladdning av opassande data inte var till så stor nytta. Detta ledde till en ny funktion på hemsidan som tillät administratörer att redigera alla restauranger.

Vi får ett förslag från en kund som tycker att det är omständigt att redigera varje meny för sig att bygga ett verktyg som läser av menyerna automatiskt. Så jag satte genast igång och efter 2 dagar har vi ett verktyg som kan skanna av en enkel meny och automatiskt lägga in rätterna. (Bilaga 11). Verktöget underlättade arbetet att lägga in en meny väsentligt och var mycket uppskattat av kunden. "Menyskannern" saknade dock vissa funktioner och några dagar senare uppdaterade jag den med möjlighet att lägga in flera rätter på samma dag genom att dela dem med ett "/". Jag lade också till möjligheten att prissätta rätter samt ge dem namn automatiskt. (Bilaga 12).



Bilaga 11 - Menyskanner, automatisk inläsning av meny.

ex.

Måndag  
Korvstroganoff med ris

Tisdag  
Gulaschgryta på högre/v  
Pasta Bolognese

Onsdag  
Grillad kycklingfilé/99

EN rätt kommer att tilldelas måndag,  
TVÅ rätter kommer att tilldelas tisdag,  
En rätt med priset 99kr kommer att tilldelas  
onsdag, priset på resten av rätterna bestäms efter  
avkodning.

Bilaga 12 - Instruktioner för menyskanner.

Efter samtal med kund som vill hantera flera restauranger möjliggör jag för moderatorer att redigera ett par stycken restauranger medan administratörer har tillgång till alla restauranger.

Jag avslutar med att skriva rapporten men fortsätter också med att bygga om apparna i Flutter vilket tillåter mig att skriva en app till båda operativsystem, detta har jag dock inte inkluderat i mitt gymnasiearbete.

## 2.2.1 PROGRAMVARA

**Programmeringsspråk:** *PHP, JavaScript, Swift, Java, HTML.*

**IDEer:** *Android Studio, Xcode, Visual Studio Code,*

**Annat:** *Trello*

### 3. RESULTATREDOVISNING

#### 3.1 SVAR PÅ FRÅGESTÄLLNINGAR

- **Hur utvecklar man en app?**

*Jag använde mig av Googles och Apples egna plattformar för att utveckla apparna, detta fungerade bra men innebar att jag var tvungen att utveckla en app till varje operativsystem. Utanför gymnasiearbetet började jag utveckla appen i Google Flutter som gör det möjligt att utveckla en app som fungerar på alla enheter.*

- **Hur ska datan hanteras?**

*Datan hanteras av Google Firebase. Detta tillät mig att säkert hantera data utan att själv behöva bygga upp alla funktioner och säkerhetsåtgärder. Det ger också möjligheten att ange speciella regler för olika "dokument" vilket används för att begränsa redigeringen av en restaurang till restaurangen själv samt administratörer medan alla andra endast kan läsa från dokumentet.*

- **Hur kan restaurangerna själva redigera sina rätter?**

*Genom ett webbaserat webbinterface har restaurangerna möjlighet att skanna in sin meny, finslipa rätternas formatering samt lägga till allergiinformation och bilder. Webbsidan serveras av Google Firebase Hosting vilket även ger ett gratis HTTPS protokoll för säkra anslutningar.*

## 4. DISKUSSION OCH ANALYS

### 4.1 FÖRBÄTTRINGAR

Jag är väldigt nöjd med hemsidan och appen på IOS även om det på appen saknas tillexempel sökfunktioner. Appen till android ligger dock efter IOS i utvecklingen och har inte ännu fått filtreringsfunktioner eller möjlighet att ändra dag. Utanför gymnasiearbetet har jag påbörjat projektet att förbättra appen och har då tagit tillfället i akt att bygga en helt ny app i Flutter som fungerar till både Ios och Android vilket innebär att arbetet som krävs minskar.

### 4.2 INKÖP

#### Inköp

Produkt	Summa
Google play servicekostnad för uppladdning	250 kr
Appstore servicekostnad för uppladdning	1000 kr
<b>Totalsumma:</b>	1250 kr

Butikerna för appar på Android och IOS har en kostnad för uppladdning, där android har en engångskostnad på 250kr medan Apple har en årskostnad på 1000 kr. Dessa behövdes för Android betalas vid uppladdning där den i IOS istället behövdes betalas i början för att få tillgång till utvecklingsverktygen.

Då jag har använt mig av skolans server samt Google Firebase har jag inte behövt betala för server-värd. Google Firebase har en generös prisplan för små företag där de första 50,000 läsningarna samt 20,000 ändringarna per dag är gratis. Efter detta kostar det \$0.06 för 100,000 läsningar samt \$0.18 för 100,000 ändringar. När en användare startar appen läses så många dokument som det finns restauranger tillgängliga i den staden. I det fall att 20 restauranger är anslutna och varje användare öppnar appen en gång per dag behövs 2500 användare för att överstiga den dagliga kvoten. Efter detta kostar det oss ca 3 kr/1000 användare/månad vilket är väldigt lite. I detta fallet har jag bara räknat med läsningar då det hade behövts mycket mer för antalet ändringar att överstiga 20,000st/dag. En restaurang uppdaterar sina rätter en gång i veckan och åstadkommer då en ändring.



## 4.3 PROGRAMVARA

I början visste jag inte vilka program jag skulle använda mig av utan det har jag fått ta reda på efter hand. Det har hänt många gånger att jag har bytt mellan olika program för att underlätta eller förbättra.

På server och databassidan använde jag mig först av ett enkelt verktyg i Visual Studio för att få en förståelse för hur data lagras i SQL. Jag gick sedan över till PHPmyAdmin vilket är verktyget som skolan använder på sina servrar för att redigera databaser. Detta fick jag göra för att kunna använda min struktur på skolans server. Jag använde mig av språket PHP för att bygga en API och backend som skulle kommunicera mellan servern och appen. Jag valde detta framför det alternativa språket Ruby då det verkade lättare att komma igång med.

Jag fick senare i processen behov av bättre serverstruktur och växlade till att använda Google Firebase som server-värd samt backend och API. Detta gjorde det också möjligt att bygga om några påbörjade hemsidor i JavaScript istället för PHP. Bytet öppnade också upp för möjligheten att använda Googles inbyggda verktyg. Några av deras verktyg som jag använder är följande, Cloud Firestore för att lagra data, Cloud Functions för att skicka mail på uppdaterade rätter, Authentication för säkra konton, Hosting för hemsidan, Cloud Storage för att lagra restaurangers bilder, Google Analytics för att få data från app-användarna om appen krashar eller liknande.

För att bygga apparna har jag för Android använt Android Studio och Java och för IOS har jag använt Xcode och Swift då detta är de program som Apple och Google själva har släppt som utvecklingsmiljö för att bygga appar till deras operativsystem.

När jag inte har använt mig av Android eller IOS egna IDE har jag använt mig av Visual Studio Code. Vilket då innebär att jag har använt det till all programmering som hanterar webbtjänster såsom hemsidor och serverstrukturer.

Under arbetets gång har jag aktivt lagt in nya idéer på Trello som är ett verktyg för att ordna uppgifter i projekt för att hålla koll på vad som behöver göras samt antalet timmar jag har lagt på de olika uppgifterna.

## 4.4 RAPPORTARBETE

För ett effektivt arbetssätt har jag suttit i långa perioder och få gånger. Då jag har mycket lättare att fokusera efter de första 20 - 30 minuterna är detta det bästa sättet för mig. Jag har hela tiden antecknat vad, när och hur länge jag har arbetat på Trello vilket jag under två omgångar har fört in i loggboken.

# 5.KÄLL- OCH LITTERATURFÖRTECKNING

## 5.1 WEBBKÄLLOR

Nedan står några viktiga källor och inte fullständig källförteckning, denna kan istället hittas på <https://lunchdaxuf.se/webbkällorGA>.

[1] Firebase produkter. Hämtad 31 Mar 2019 20:34:46 från <https://firebase.google.com/products/>

[2] Firebase prissättning. Hämtad 31 Oct 2018 17:21:19 från <https://firebase.google.com/pricing/?authuser=0>

[3] IOS/XCode Tutorial: A Card-based Newsfeed – cw Richard Kim. Hämtad 21 Nov 2018 17:50:32 från <https://blog.cwrichardkim.com/ios-xcode-tutorial-a-card-based-newsfeed-8bedeb7b8df7>

[4] Firebase Tutorial: Getting Started | raywenderlich.com. Hämtad 23 Oct 2018 17:39:56 från <https://www.raywenderlich.com/3-firebase-tutorial-getting-started>

[5] SQL Tutorial. Hämtad 3 Oct 2018 14:6:15 från <https://www.w3schools.com/sql/>

[6] Google maps places API pricing. Hämtad 20 Apr 2019 13:05:40 från <https://developers.google.com/places/web-service/usage-and-billing>

[7] GitHub - xmartlabs/Eureka: Elegant iOS form builder in Swift. Hämtad 1 Apr 2019 11:5:7 från <https://github.com/xmartlabs/Eureka>

[8] The Firebase Blog: How to Schedule (Cron) Jobs with Cloud Functions for Firebase. Hämtad 7 Apr 2019 13:42:41 från <https://firebase.googleblog.com/2017/03/how-to-schedule-cron-jobs-with-cloud.html?m=1>

[9] Writing conditions for Cloud Firestore Security Rules | Firebase. Hämtad 6 Apr 2019 22:31:9 från <https://firebase.google.com/docs/firestore/security/rules-conditions>

[10] Secure Your Data | Firebase Realtime Database | Firebase. Hämtad 6 Apr 2019 22:22:50 från [https://firebase.google.com/docs/database/security/securing-data#validating\\_data](https://firebase.google.com/docs/database/security/securing-data#validating_data)

[11] User Based Security | Firebase Realtime Database | Firebase. Hämtad 6 Apr 2019  
22:22:45 från <https://firebase.google.com/docs/database/security/user-security>

[12] risk of javascript injection attacks with firebase - Stack Overflow. Hämtad 6 Apr 2019  
22:20:32 från <https://stackoverflow.com/questions/36254539/risk-of-javascript-injection-attacks-with-firebase>

## 5.2 VIDEOKÄLLOR

Nedan står några viktiga videokällor och inte fullständig källförteckning, denna kan istället hittas på <https://lunchdaxuf.se/videokällorGA>.

[1] Firestore Security Rules - How to Hack a Firebase App - YouTube. Hämtad 11 Nov 2018  
17:44:20 från <https://www.youtube.com/watch?v=b7PUm7LmAOW>

[2] Firebase Database Rules Tutorial - YouTube. Hämtad 11 Nov 2018  
11:19:37 från <https://www.youtube.com/watch?v=qLrDWBKTUzo>

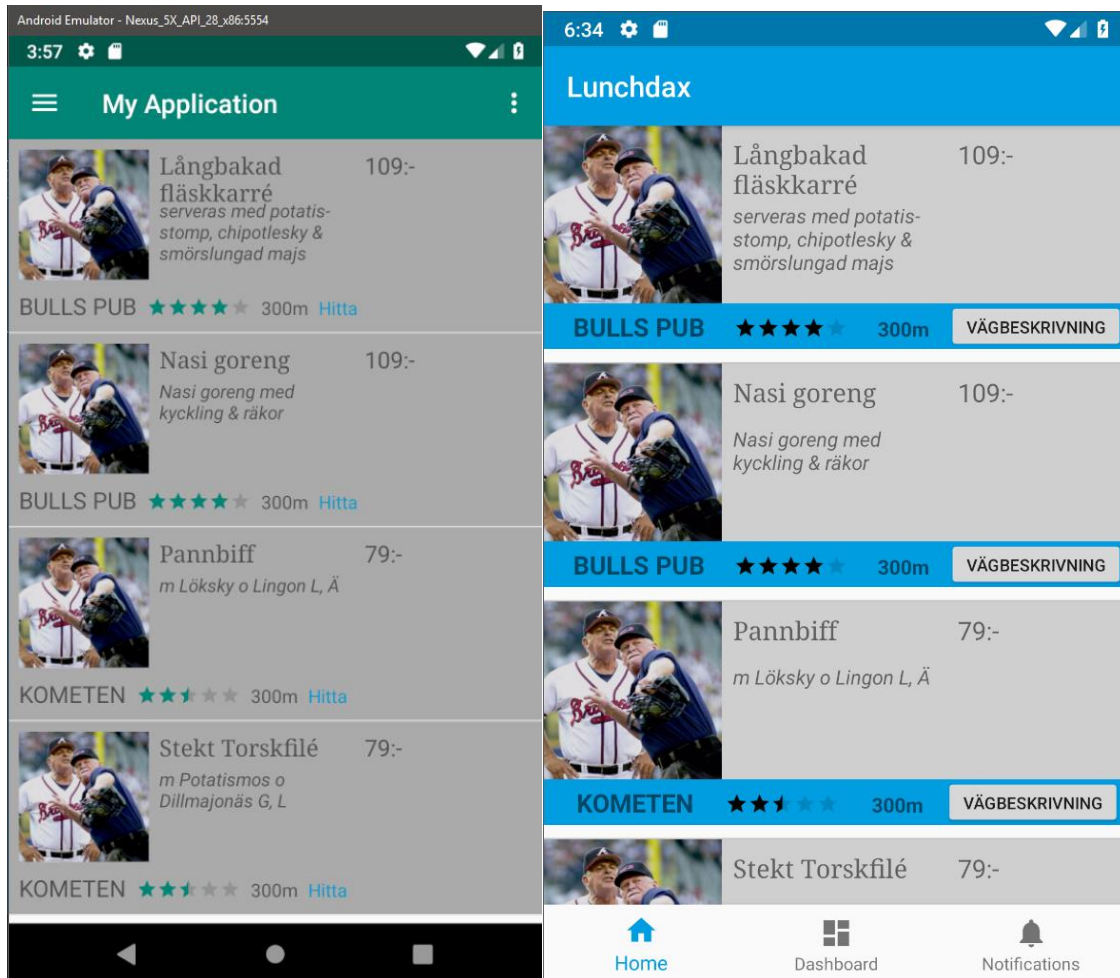
[3] Getting Started with Firebase Storage on the Web - Firecasts - YouTube. Hämtad 8 Nov 2018  
16:56:14 från <https://www.youtube.com/watch?v=SpxHVrpfGgU&t=6s>

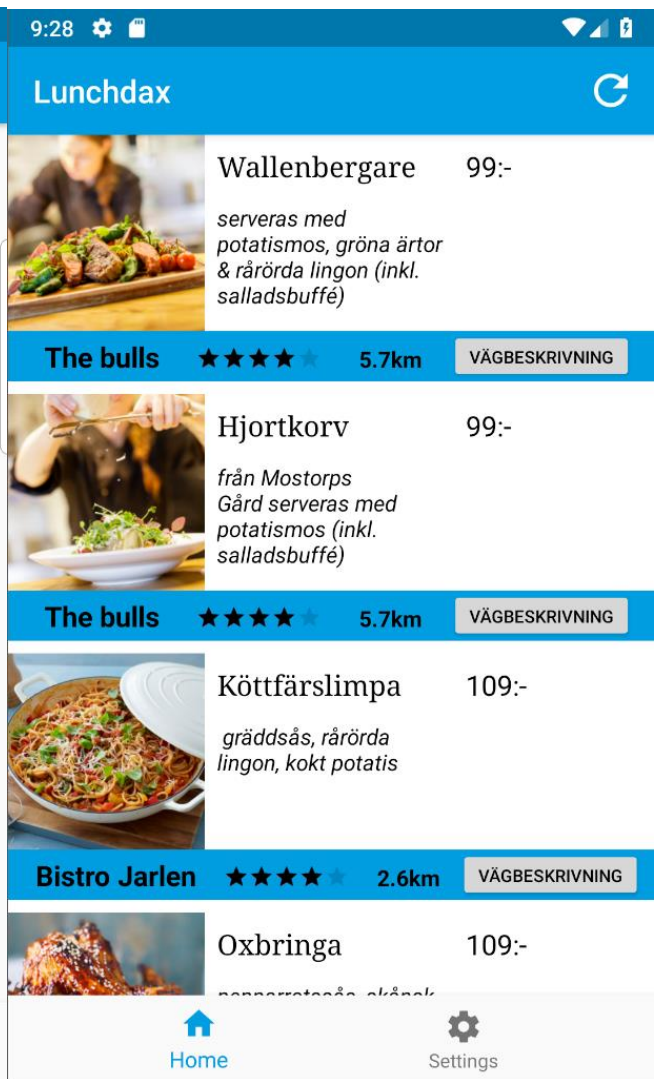
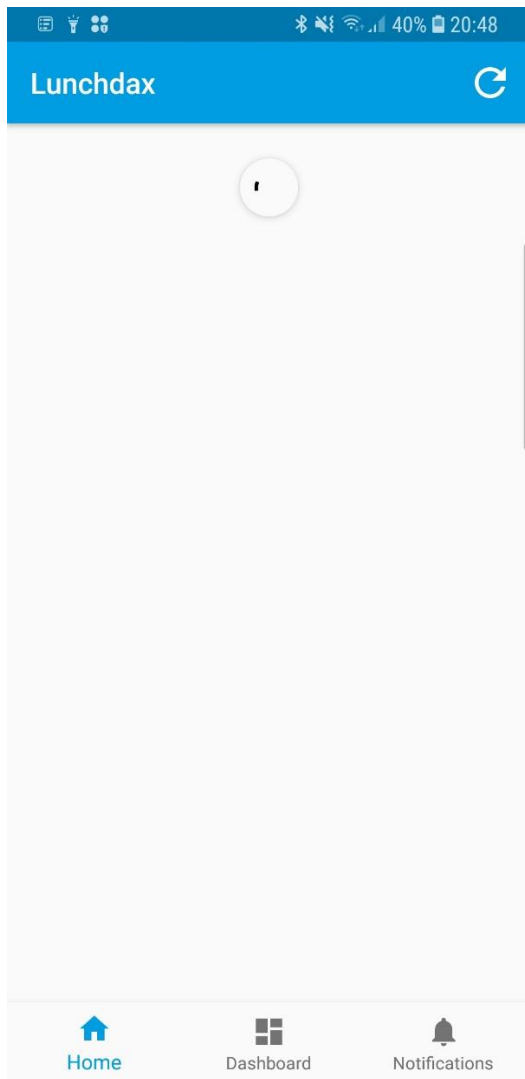
[4] Make a Simple Android App - Xamarin Android Tutorial - YouTube. Hämtad 23 Sep 2018  
18:5:13 från <https://www.youtube.com/watch?v=5CgQUbnf1Qk>

[5] Card View in iOS (Xcode8 , Swift3) - YouTube. Hämtad 21 Nov 2018  
17:53:35 från <https://www.youtube.com/watch?v=y-Tk7biMIOM>

## 6. BILAGOR

### 6.1 APP ANDROID UTVECKLING






23%

14:18

Lunchdax



Omelett

99:-


*Omelett ostronskivling, serveras med bacon*

Fungigården

★★★★★

1.7km

KARTA



Paj

109:-


*OstronskivlingPaj serveras med krämig sås*

Fungigården

★★★★★

1.7km

KARTA



Anka

139:-

*Ostronskivling i sesambuljong med anka*

Fungigården

★★★★★

1.7km

KARTA



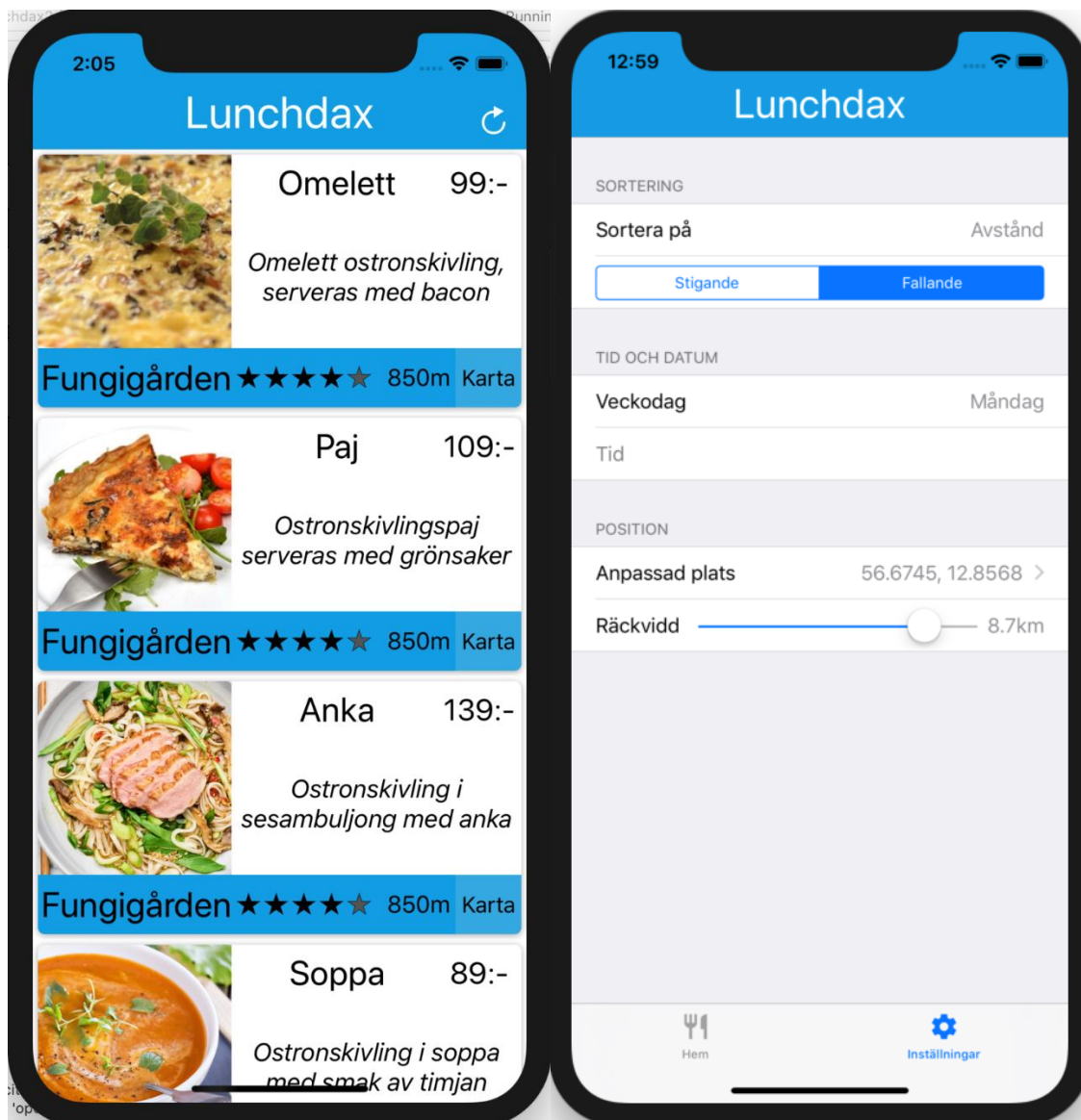
Soppa

89:-

Home

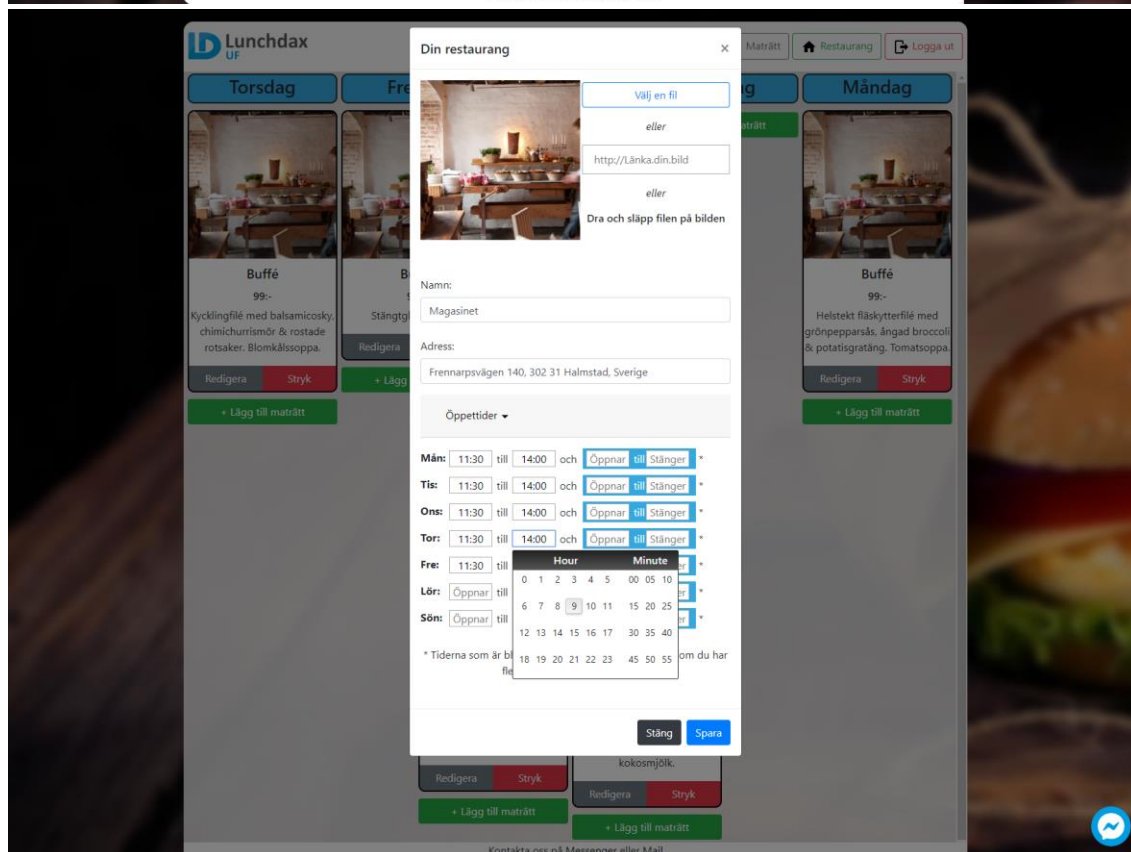
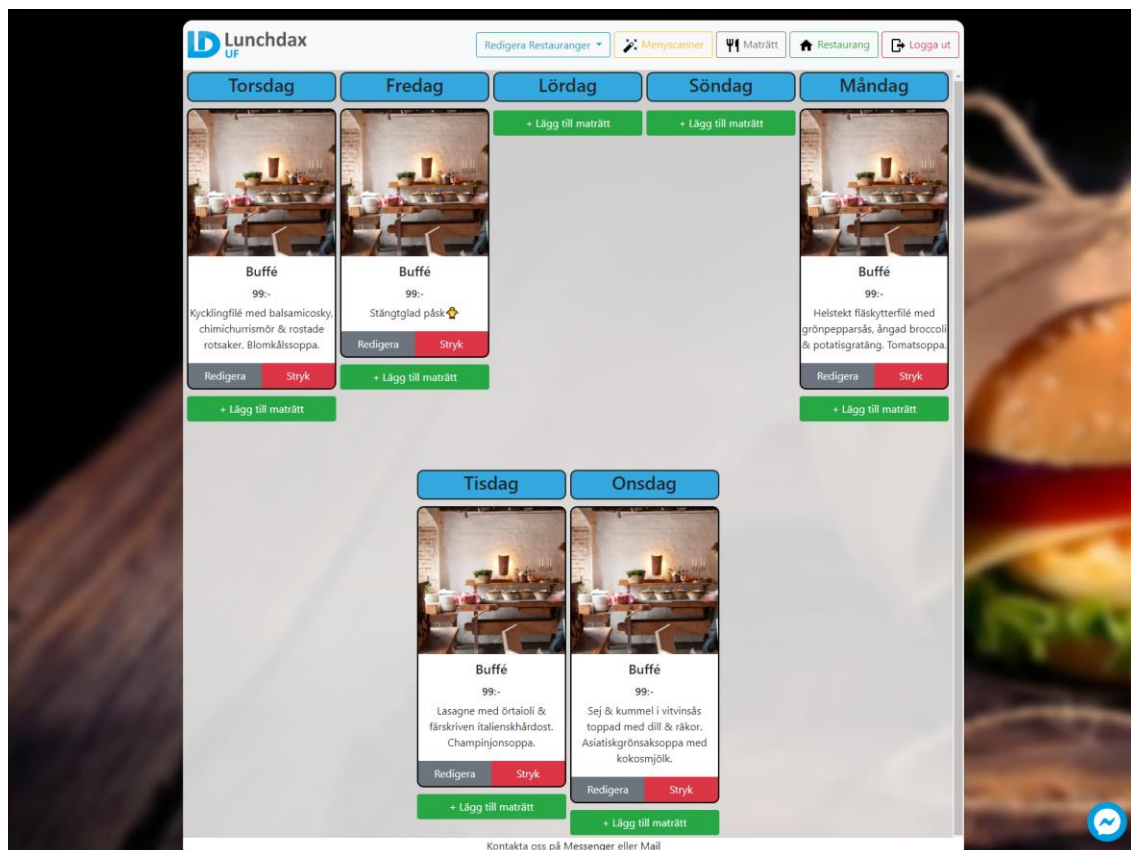
Settings

## 6.2 APP IOS





## 6.3 HEMSIDA





## 6.4 VIDEOR HEMSIDA

**Återställning av lösenord:**

[https://trello-attachments.s3.amazonaws.com/5baa4e7a3886484aae8db774/5bdc7f925c854d8b67b3b70b/377b926dbc4f5b4377a0cc27c74c0290/lunchdax\\_%C3%A5terst%C3%A4ll\\_l%C3%B6senord.gif](https://trello-attachments.s3.amazonaws.com/5baa4e7a3886484aae8db774/5bdc7f925c854d8b67b3b70b/377b926dbc4f5b4377a0cc27c74c0290/lunchdax_%C3%A5terst%C3%A4ll_l%C3%B6senord.gif)

**Skapar ett konto till en restaurang, i slutet visas den data som lades in i databasen, öppettider, position, adress mm:**

[https://trello-attachments.s3.amazonaws.com/5baa4e7a3886484aae8db774/5bde3150fee53273a4f56dc9/72a631a56f2527ea59889308c95b82c3/lunchdax\\_skapa\\_konto.gif](https://trello-attachments.s3.amazonaws.com/5baa4e7a3886484aae8db774/5bde3150fee53273a4f56dc9/72a631a56f2527ea59889308c95b82c3/lunchdax_skapa_konto.gif)

## 6.5 LOGGBOK

*Dubbelklicka på filen för att öppna loggboken, fungerar endast i word*



Loggbok.xlsx