



Bokningssystem för Kårspexet

User Requirements Document (URD)

Tarandi, Andreas – taran@kth.se
Arvidsson, Kalle – kallear@kth.se
Boström, Peter – pbos@kth.se
Eklund, Erik – eekl@kth.se
Gräsman, André – grasman@kth.se
Göransson, Rasmus – rasmusgo@kth.se
Hagsten, Per – hagsten@kth.se
Hallberg, Victor – victorha@kth.se
Modée, Anna Maria – ammodee@kth.se
Nyberg, Daniel – dnyb@kth.se
Stjernberg, Johan – stjer@kth.se

Nyx

Version 0.2

December 1, 2010

Abstract

Genererat från revision 3 av "URDabstract".

This document aims to describe the user requirements for Kårspexets online booking system, developed by Nyx. It contains the systems capabilities, its constraints, assumptions, and dependencies, as well as its user characteristics, and operational environment. The document also features an extensive list of detailed requirements of the system, deduced from the preceding descriptions.

Innehåll

1	Ändringslogg	1
2	Introduktion	2
2.1	Syfte	2
2.2	Mjukvarans omfattning	2
2.3	Definitioner, akronymer och förkortningar	2
2.4	Källor	5
2.5	Dokumentöversikt	5
3	Allmän beskrivning	6
3.1	Produktperspektiv	6
3.2	“General Capabilities”	6
3.3	Allmänna begränsningar	8
3.4	Användarbeskrivning	9
3.5	Antaganden och beroenden	11
3.6	“Operational environment”	12
4	Specifika krav	14
4.1	Begränsande krav	20

1 Ändringslogg

Genererat från revision 4 av "URDÄndringslogg".

Version	Ändringar
0.1	Första sammanställd version av dokumentet.
0.2	Texter under bearbetning men i stort sätt färdigkomponerade. Texter i dokumentet numreras numera med sitt revisionsnummer i dokumenthanteringssystemet för spårbarhet. Kravdatabas innehåller kundens funktionalitetskrav, men beskrivande texter saknas till vissa av dem.

2 Introduktion

Genererat från revision 4 av “URDIntroduktion”.

2.1 Syfte

Genererat från revision 5 av “URDIntroduktionSyfte”.

Det här dokumentet innehåller en detaljerad sammanfattning av produktens krav och dess motiveringar inom vårt projekt. Dokumentets tänkta läsare är tekniskt datakunniga personer samt utvecklare.

2.2 Mjukvarans omfattning

Genererat från revision 3 av “URDIntroduktionMjukvarans__omfattning”.

Produkten består av ett webbaserat biljettbokningssystem med ett enkelt användargränssnitt och ett kraftfullt administrationsverktyg. Administrationsgränssnittet utgörs av tre delar; ett för säljare, ett för ekonomiansvariga och ett för administratörer.

2.3 Definitioner, akronymer och förkortningar

Genererat från revision 42 av “URDIntroduktionDefinitioner__akronymer__och__förkortningar”.

Algoritm Inom matematik och datorvetenskap är detta en begränsad uppsättning tydliga instruktioner för att utföra en uppgift.

Apache I datasammanhang en webserver som är gratis att använda.

Applikation I datasammanhang även kallat tillämpningsprogram. Ett dataprogram som fyller ett direkt syfte för användaren.

Bandbredd I vardagligt tal en storhet för hur mycket information som kan överföras på en viss tid. Vanlig enhet är Mbit/sekund.

Bit (Binary Digit) Den grundläggande enhet som datorer arbetar med. En bit kan anta ett utav två möjliga värden (ofta angivna som 0 eller 1).

Byte En vanlig enhet för informationsmängd i datasammanhang. En byte är ett paket bestående av åtta bitar.

CentOS Ett operativsystem baserat på Red Hat Enterprise Linux som är gratis att använda.

Databas En databas är en samling information ordnad på ett sådant sätt att informationen i den effektivt går att hitta.

Epostklient Även kallat Epostprogram. Datorprogram för att hantera/läsa/skicka epost.

Epostprogram Se Epostklient.

FK *Foreign Key*, ett fält som innehåller ett värde som är sparad i en annan tabell

GHz *Enhet för antalet miljarder svängningar per sekund. "G" är binärt prefix för 10^9 . "Hz" är förkortning för Hertz.*

Gränssnitt *Utformningen av en mjukvarumodul avsedd att användas av användare eller annan mjukvara för att kommunicera med hårdvarumoduler.*

HTML (Hyper Text Markup Language) *Ett språk och webbstandard som används för att beskriva strukturering av text, bilder och annan media på en webbsida.*

HTTP (HyperText Transfer Protocol) *Ett standardiserat protokoll som definierar hur kommunikation över webben sker.*

Hårdvara *Även kallat Maskinvara. Ett samlingsnamn för en dators fysiska komponenter.*

kB (kilobyte) *Se kbyte.*

kbyte (kilobyte) *Enhet för datamängd. "k" är prefix för 10^3 . För "byte", se Byte.*

KiB (kibibyte) *Enhet för datamängd. "Ki" är ett binärt prefix för 2^{10} . "B" är förkortning för Byte.*

Klockfrekvens *Beteckning för den hastighet i vilken en processor arbetar i.*

KTH (Kungliga Tekniska Högskolan) *Sveriges största tekniska universitet.*

Latens *Även känt som svarstid, tidsfördröjning eller lagg. Tidsskillnaden mellan en begäran och respons på begäran.*

Latency *Engelskt ord för Latens.*

Linux *Open source Unix liknande operativsystem.*

Mb (Megabyte) *Se Mbyte.*

Mbyte (Megabyte) *Enhet för datamängd. "M" är prefix för 10^6 . För "byte", se Byte.*

MHz *Enhet för antalet miljoner svängningar per sekund. "M" är binärt prefix för 10^6 . "Hz" är förkortning för Hertz.*

MiB (mebibyte) *Enhet för datamängd. "Mi" är ett binärt prefix för 2^{20} . "B" är förkortning för Byte.*

MiBit/s (mebibit per sekund) *Enhet för datahastighet. "Mi" är ett binärt prefix för 2^{20} . "Bit" är den minsta enheten för informationsmängder i datasammanhang.*

Mjukvara *Även kallat programvara. En organiserad samling av data och maskininstruktioner.*

Mjukvarubibliotek *En samling av redan existerande program eller delar av program som används för att utveckla mjukvara.*

MVC (Model-View-Controller) *Ett koncept som bygger på att separera data (modeller), logik (kontroller) och användarinterface (vyer).*

MySQL *En typ av relationsdatabas baserad på SQL-standard. Ett relationsdatabas hante-*

ringssystem där flera användare kan arbeta med flera databaser.

Open Source *Engelskt låneord för öppen källkod.*

Operativsystem *Ett datorprogram vars syfte är att underlätta användandet av en dator genom att vara länken mellan programvara och hårdvara.*

PHP *Ett programmeringsspråk som ofta används för att skapa webbapplikationer.*

PK *Primary Key, det/de fält som gör en rad i tabellen unik*

Processor *Den komponent i en dator som genomför beräkningar och andra instruktioner.*

Programmeringsspråk *Språk som människor använder för att skapa datorprogram.*

Rails *I datorsammanhang vanlig förkortning för Ruby on Rails.*

Red Hat Enterprise Linux *Variant av Linux.*

Ruby *Ett objektorienterat programmeringsspråk.*

Ruby on Rails *Ett abstrakt mjukvarubibliotek med öppen källkod för utveckling av webbapplikationer.*

Systemminne *Även kallat RAM (Random Access Memory) eller Arbetsminne. Utgör datorns primärminne och används för att tillfälligt lagra data som datorn arbetar med.*

Spex *Humoristisk studentamatörteaterföreställning, förkortning av spektakel.*

SQL (Structured Query Language) *Ett språk designat för att interagera med databaser.*

Unix *Ett operativsystem som ofta används i olika typer av servrar och arbetsstationer.*

URL (Uniform Resource Locator) *Den formella benämningen av en webbadress. En text som beskriver var en viss resurs på internet finns, samt hur den går att komma åt.*

Webb *Även känt som WWW (World Wide Web). Det system som används för att hämta, visa och manipulera delar på internet. WWW utgörs av standarderna URL, HTTP respektive HTML.*

Webbapplikation *Samlingsnamn för mjukvara som användare kommer åt via en webbläsare.*

Webbläsare *Ett program som hämtar, tolkar och återger webbsidor kodade exempelvis HTML.*

Webbserver *Program som körs på en server och distribuerar webbsidor och/eller andra filer som en webbläsare begär via HTTP-protokollet.*

Webbsida *En fil, innehållandes exempelvis HTML, avsedd att visas av en webbläsare.*

Öppen källkod *Innebär möjlighet att ändra i konstruktionen för ett system. I ett datorprogram som har öppen källkod kan den som vill göra ändringar i programmet och utveckla det vidare.*

2.4 Källor

Genererat från revision 2 av “URDIntroduktionKällor”.

2.5 Dokumentöversikt

Genererat från revision 3 av “URDIntroduktionDokumentöversikt”.

Provides a birds-eye view of what information is given in this report, and where in the report it can be found. Description can be focused towards different types of reader, e.g. end-user, technical, developer, specialist, domain expert, accountant, legal, management, customers customer etc.

3 Allmän beskrivning

Genererat från revision 2 av "URDAllmän_beskrivning".

3.1 Produktperspektiv

Genererat från revision 19 av "URDAllmän_beskrivningProduktperspektiv".

Kårspexet vill ha ett nytt bokningssystem till sina föreställningar eftersom de är missnöjda med sin nuvarande lösning. De vill ha ett väldokumenterat system med tillgång till källkoden så de kan vidareutveckla systemet vid behov. Lösningen måste vara enkel så att Kårspexet slipper lägga mer tid än nödvändigt på administrationen. På så sätt får de mer tid att fokusera på andra aktiviteter som marknadsföring och anordna bra spex.

Vårt uppdrag är att skapa ett nytt bokningssystem efter Kårspexets önskemål. Vi skall fokusera på att skapa ett enkelt och visuellt tilltalande system för Kårspexet och deras besökare. Systemet som används idag är tråkigt, något föråldrat och ser komplicerat ut. Det system som vi kommer att konstruera skall ha olika vyer för kunder, administratörer och säljare. Varje vy kommer att anpassas för sin målgrupp och dokumenteras där efter. På så sätt kommer interaktionen mot bokningssystemet bli lättare och angenämare för användarna.

Ett lättare och angenämare bokningssystem kan hjälpa Kårspexet att höja sina intäkter genom en ökad biljettförsäljning. Svåra och icke användarvänliga system kan få osäkra besökare att avstå från ett köp, medans ett enkelt system kan locka till sig fler kunder. Ett bra bokningssystem kan ge ett bättre intryck inför studenter och andra besökare. Vilket kan ge möjligheten att producera fler spex som leder till ytterligare intäkter.

// André ändrat lite till.. Nu så:

Cruella de vil has left the building!

Vi stampar lite försiktigare på fysikalen

3.2 "General Capabilities"

Genererat från revision 5 av "URDAllmän_beskrivningGeneral_Capabilities".

Bokningssystemet ska användas av fyra typer av användare: kund, säljare, ekonomichef och administratör. Dessa har olika roller som interagerar med varandra.

3.2.1 Boka biljetter från hemsidan

Kunder ska kunna boka biljetter från kårspexets hemsida. Efter att kunden har genomfört en bokning ska kunden få ett mail med betalningsuppgifter och bokningsnummer.

3.2.2 Registrera betalningar

Ekonomichefen ska kunna registrera betalningar för bokningar som kunder gjort.

3.2.3 Administrera biljetter

Efter att kunden har bokat och ekomichefen registrerat kundens betalning placerar administratören ut vilka stolar kunden ska få sitta på under föreställningen. När placeringen är klar får kunden ett mail som uppmanar kunden att hämta ut sina biljetter.

3.2.4 Lämna ut bokade biljetter

När kunden kommer till säljaren ska denna verifiera att biljetterna är redo att hämtas samt registrera att biljetterna har hämtats. Gränssnittet för säljaren ska vara lätt att lära sig eftersom säljarna ofta byts ut.

3.2.5 Sälja biljetter direkt

Säljaren ska även kunna sälja biljetter kontant. Då är det säljaren som väljer vilka stolar som kunden får sitta på.

3.2.6 Administrera mailutskick

Administratören ska kunna ändra informationen i de automatiska utskicken som sker samt kunna göra nya utskick till valda kunder.

3.2.7 Administrera föreställningar och teatrar

När det vankas nya föreställningar är det administratören som matar in dem i systemet. Priser ska kunna ändras och om föreställningen är på en ny teater ska den kunna läggas till. Detta innebär att nya salongsskisser med nya sektioner och stolar ska kunna matas in. Administratören ska kunna välja vilka föreställningar det går att boka/köpa biljetter till.

3.2.8 Administrera konton

Administratören ska kunna ändra både sitt eget och andras lösenord. Säljarens lösenord ska kunna genereras automatiskt och vara giltigt en begränsad tid.

3.2.9 Statistik

Administratören och ekomichefen ska kunna se statistik om antalet bokade och sålda biljetter för att kunna få inblick i verksamheten.

3.3 Allmänna begränsningar

Genererat från revision 27 av "URDAllmän_beskrivningAllmänna_begränsningar".

General constraints. Describes the main constraints that apply to the product, and why they exist.

3.3.1 Datamodell

Describes the constraints on data in the form of a data model, which may be an object diagram, class diagram or data dictionary. (Note: a data model must be presented.)

3.3.2 Resurser

Vi kommer vara begränsade i vilka och hur många funktioner vi kommer kunna implementera främst då vi totalt är fem programmerare. På tre veckor ska vi hinna implementera fyra gränssnitt mot bokningssystemets användare. Gränssnitten kommer behöva testas men då vi har nästan lika många testare som vi har utvecklare kommer inte detta utgöra ett hinder för tidsplanen. Det som kommer vara vårt största hinder under utvecklingen är administratörsgränssnittet då det är där de flesta funktionerna kommer finnas och även de mest avancerade.

Vi har inte någon budget för projektet och vi kommer inte att tillföra egna pengar för att köpa in något, detta gör att vi begränsas till att använda programvara som är gratis. Detta skulle kunna innebära ett problem i vissa projekt, men just inom webbutveckling finns det starka open source-programvaror att använda för våra ändamål.

3.3.3 Kundbehov

Nyx mål är att leverera ett fullständigt bokningssystem med alla de funktioner som Kårspexet har specificerat. På grund av systemets förväntade komplexitet och tidsramen vi har kommer kvaliteten i delar av slutprodukten vara begränsad.

Gränssnittet för besökare (slutkunder) respektive säljare ska designas på ett sätt som gör att det går att använda utan några speciella förkunskaper inom vårt system. Det ska alltså fungera på ett sätt som efterliknar liknande produkter. Detta begränsar oss i hur pass många funktioner och val vi kan låta användarna exponeras för på en och samma gång. Administrationsgränssnittet är inte begränsat på samma sätt då dess användare kommer utbildas i förväg.

3.3.4 Tekniska begränsningar

Vi är begränsade till att utnyttja mjukvara som är fri att använda, dels på grund av avsaknad av ekonomiska resurser, och dels för att vi valt detta redan från början. (*står om detta i början, bör ev. skrivas om?*)

Kårspexet står för den server som kommer att köra vår webbapplikation. Vi har ingen kontroll över dess hårdvara men har verifierat att operativsystemet som körs på den är kompatibelt med

Apache, Ruby med Rails [1] samt MySQL.

Applikationen kommer inte att inkludera avancerade algoritmer utan till störst del involverade mycket trafik till och från databasen. I och med att webbapplikationen och databasen körs på en och samma dator undviks eventuella begränsningar i kabelanslutningar.

Systemet kommer enligt våra uppskattningar exponeras för upp till tio simultanta anslutningar. Rails under Apache kommer i detta fall att, under godtycklig tidpunkt, använda uppskattningsvis 250 mb systemminne [2]. CentOS anger 256 mb minne samt en klockfrekvens på minst 500 MHz som minimikrav för datorer som kör operativsystemet [3]. Med MySQL och Apache körandes samtidigt utöver dessa bör servern ha minst en gigabyte systemminne samt en processor med klockfrekvensen 1 GHz eller högre.

LÄGG IN FAKTISK SPEC FÖR KÅRSPEXETS SERVER HÄR

[1]: <http://hasham2.blogspot.com/2008/07/install-phusion-passenger-on-cent-os-5.html>

[2]: <http://www.rubyenterpriseedition.com/comparisons.html>

[3]: http://www.centos.org/docs/5/html/CDS/install/8.0/Installation_Guide-Support-Platforms.html

3.4 Användarbeskrivning

Genererat från revision 23 av "URDAllmän_beskrivningAnvändarbeskrivning".

Produkten kommer att ha 4 olika användare: Kund, säljare, administratör och ekonomiansvarig.

3.4.1 Kund

Teknisk bakgrund för kund Kunder är uppdelad i två läger; de som pluggar på en teknisk högskola eller ett universitet, och släktingar eller bekanta till Kårspexets medlemmar som inte är associerade med en teknisk högskola eller ett universitet.

Tekniska högskolestudenter Använder datorer dagligen, antingen som del av sin utbildning och/eller för privat bruk. Van användare av emailklienter, och anpassar sig i behaglig takt till nya sidlayouter eller program.

Släktingar och bekanta Den tekniska kunnigheten varierar stort inom denna grupp, från datorvana tonåringar till pensionärer som aldrig rört en dator. Kan använda emailklienter till viss mån, och tar lång tid på sig att anpassa sig till nya sidlayouter eller program.

Typscenario för kunden Kunden går in på Kårspexets hemsida, och trycker på boka biljett. Kunden behöver en snabb och genomförlig överblick av vilka föreställningar som finns, och hur många platser som finns i respektive sektioner. Kunden får snabbt återkoppling på sina val och går igenom flera steg av bokningen; val av föreställning, val av sektion, betalningsuppgifter, och

bokningsbekräftelse. Kunden får sedan ett mail från Kårspexet om betalmedel, och betalar sin biljett. Kunden får sedan ett mail från Kårspexet om att hans/hennes biljett finns att hämta hos ombud.

3.4.2 Säljare

Teknisk bakgrund för säljare Säljare är medlemmer i Kårspexet, är därmed kårmedlemmar vid en teknisk högskola. De är först och främst aktiva med kårspexet arrangemang, och säljare i andra hand. Det är därför viktigt att säljare snabbt lär sig använda bokningssystemet, då de inte ska kräva någon tidigare utbildning i systemet.

Typscenario för säljare Säljaren står i Kårhuset eller på utsatt plats, och loggar in på Kårspexets hemsida. Antingen så säljs biljetter på plats, med kunder som står i kö; eller så kommer en kund som bokad sin biljett på hemsidan och valt att betala kontant. I båda fallen så placerar säljaren ut en plats i den sektion kunden har valt, och tar emot betalning för bokningen. Säljaren ger även ut utplacerade biljetter som blivit betalda, till kunder som fått mail från Kårspexet om att deras biljett finns att hämta.

3.4.3 Administratör och ekonomiansvarig

Allmän teknisk bakgrund för administratören och ekonomiansvarige Både administratören och ekonomiansvarige är studenter på en teknisk högskola, i Kårspexets fall KTH. De är därmed vana att navigera personliga inloggningssidor, t.ex. Mina sidor, eller studera.nu. De är även vana användare av emailklienter. Deras tekniska bakgrund är uppdelad i två ganska jämna läger; de mindre tekniskt kunniga, och de med lite mer teknisk bakgrund.

Mindre teknisk kunnig Organisatören kan ha en bakgrund i matematik, biologi, kemi, eller liknande ämne, där programmering inte är en väsentlig del av utbildningen. Organisatören kan hantera textredigerare väl, då han/hon är van att skriva rapporter. Det tar lite längre tid för den mindre tekniskt kunnige att använda nya program eller anpassa sig till nya sidlayouter.

Mer teknisk kunnig Organisatören kan ha en bakgrund i datalogi, teknisk fysik, eller liknande ämne, där programmering har varit del av utbildningen. Organisatören förstår objektorientering, och kanske html eller webbprogrammering. En sida söks igenom systematiskt, och organisatören lär sig snabbt nya sidlayouter eller program.

Typscenario för administratören Administratören loggar in på Kårspexets hemsida. Han/hon har uppskattningsvis 15 minuter till 2 timmar till förfogande att jobba med sina uppgifter.

Gör ofta: placera ut betalda bokningar, planera föreställningar och ta hand om specialbokningar, t.ex. handikappsbokningar eller stora företagsbokningar.

Gör mer sällan: lägga till en ny teater, lägga till en ny omgång, skicka massutskick till b.l.a. kundbokningar, kolla på statistik.

Typscenario för ekonomiansvarige Ekonomiansvarige loggar in på Kårspexets hemsida. Han/hon har uppskattningsvis 15 minuter till 2 timmar till förfogande att jobba med sina uppgifter.

Gör ofta: bockar av betalade bokningar, skickar påminnelser till obetalda bokningar, tar bort gamla bokningar.

Gör mer sällan: Kolla på utförlig statistik.

3.5 Antaganden och beroenden

Genererat från revision 20 av "URDAllmän_beskrivningAntaganden_och_beroenden".

Bokningssystemet som utvecklas för Kårspexet är beroende av datorkraft från webbservrar där mjukvaran körs. Mjukvaran och systemet i sin helhet ställer krav på yttre faktorer för att systemet skall bli användbart. De yttre faktorerna är framför allt bandbreddsuppkoppling och serverprestanda.

Bandbreddsuppkopplingen talar om i vilken hastighet webbservern kan kommunicera med omvärlden. Omvärlden som består av ett flertal användare kräver var och en, en viss del av den totala bandbredden då en användare är aktiv. Med andra ord beror behovet av bandbreddsuppkoppling på hur många som använder systemet samtidigt.

Serverprestanda talar om hur många anrop till ett system som en server kan hantera samtidigt. Varje aktiv användare kräver en del av den totala prestanda som finns tillgänglig. Behovet på serverprestanda beror precis som bandbreddsuppkopplingen på hur många som använder systemet vid samma tidpunkt.

Antalet samtidiga användare beror på en rad olika antaganden om systemet och användandet av systemet. Utifrån antagandena vill vi bestämma hur mycket prestanda och bandbredd som systemet maximalt kan kräva. De avgörande antagandena berör:

- A. Hur många platser en föreställning har i medeltal.
- B. Hur många föreställningar som släpps för biljettköp åt gången.
- C. Hur stor del av platserna som säljs per tidsenhet då efterfrågan är som störst.
- D. Hur många anrop (sidladdningar)det krävs från bokningsgränssnittet för användaren till servern under en bokning i medeltal (första inladdningen utesluten).
- E. På vilken tid antalet anrop är fördelade vid en bokning (hur lång tid det tar att boka).
- F. Hur mycket trafik som överförs vid första inladdningen av bokningsgränssnittet för användaren.
- G. Hur mycket trafik som överförs vid ett anrop (första inladdningen utesluten) i medeltal.
- H. Hur många platser som bokas vid en bokning i medeltal.

Vi antar att:

- a. att en föreställning inte har mer än 800 platser.
- b. att biljettsläpp inte görs för mer än 4 föreställningar i taget.
- c. att efterfrågan är maximalt 30
- d. att bokningsgränssnittet för användaren inte behöver anropa servern mer än 10 gånger per bokning (första inladdningen ej inräknad).
- e. att en bokning tar 4 minuter och att bokningens anrop till servern är jämt fördelat över tiden.
- f. att trafiken vid första inladdningen av bokningsgränssnittet för användaren är 100KiB.
- g. att trafiken för ett anrop (första inladdningen utesluten) är 30KiB stort i medeltal.
- h. att varje bokning omfattar 2 platser i medeltal.

Våra antaganden ger:

0,046 (anrop/sekund) för varje bokning under den tid det tar att boka $((d+1)/(e*60))$.

3200 bokningsbara platser vid varje biljettsläpp $(a*b)$.

0,27 (platser/sekund) som hanteras då efterfrågan är maximal $((a*b*c)/(100*60*60))$.

0,14 (bokningar/sekund) som hanteras då efterfrågan är maximal $((a*b*c)/(100*60*60*h))$.

1,47 (anrop/sekund) till servern då efterfrågan är maximal $((a*b*c*(d+1))/(100*60*60*h))$.

0,42 (MiBit/sekund) i trafik då efterfrågan är maximal $((a*b*c*(d*g+f)*8)/(100*60*60*h*1024))$.

Utifrån antagandena så skall bandbreddsuppkopplingen minst vara 0,42 MiBit/sekund och webbservern måste klara av att hantera 1,47 anrop/sekund. Vad gäller bandbreddsuppkopplingen så motsvarar 0,42 MiBit/sekund en mindre del av en vanlig uppkoppling i hemmet. Det låga antalet 1,47 anrop/sekund mot bokningssystemet gör att prestanda från en vanlig persondator räcker till.

Antagandena om användandet av systemet har diskuterats med Kårspexet. De antaganden som gjorts är väl tilltagna gentemot Kårspexets uppfattning av användandet. Antagandena är tilltagna på ett sådant sätt att kraven för bandbreddsuppkoppling och serverprestanda blir större. Med andra ord kommer Kårspexets användande av systemet ha lägre krav på den befintliga hårdvaran än med angivna antagandena ovan.

3.6 “Operational environment”

Genererat från revision 4 av “URDAllmän_beskrivningOperational_environment”.

Biljettsystemet kommer använda flera externa system. Till att börja med kommer ett MVC-framework för webbapplikationer i ruby, Rails att användas. Det ger oss funktionalitet som gör webbutveckling smidig, databashantering, rendering av html och mycket hjälpfunktionalitet som underlättar webbutveckling.

Vi kommer även att använda databasmotorn MySQL för datalagring. Databasen görs tillgänglig

för systemet med hjälp av SQL. Dock kommer Rails att sköta mycket av den kommunikationen åt oss och i slutändan kommer databasen vara tillgänglig genom modeller i form av klasser i koden.

Systemet kommer dessutom vara beroende av Apache 2 med modulen passenger för att sköta inladdning av applikationen och all http-kommunikation i produktion. Interfacet mot Apache är i form av konfigurationsfiler på servern.

Om vi väljer att implementera kortbetalning i systemet kommer vi även att vara beroende av ett extern system för hantering av korttransaktioner. Hur interfacet mot det systemet ser ut vet vi inte i dagsläget, eftersom inga beslut har tagits angående vilket system som ska användas.

4 Specifika krav

Genererat från revision 2 av "URDSpecifika_krav".

Kravnummer	1
Krav	Loginsystem
Beskrivning	För administration av bokningar och dylikt krävs det någon form av säkerhetssystem som hanterar användare och inloggningar. Användare identifierar sig via ett användarnamn med tillhörande lösenord.
Motivering	Av säkerhetsskäl
Behov	Standard
Prioritet	High
Källa	Kund
Verifierbarhet	Förklara hur Karl ska kolla att (1) kravet finns i designen och (2) att mjukvaran implementerar kravet.

Kravnummer	2
Krav	Boka
Beskrivning	Det ska gå att (som antingen kund eller administratör) genomföra en bokning med valfritt antal biljetter för en vald föreställning.
Motivering	Huvudfunktion i systemet
Behov	Standard
Prioritet	High
Källa	Kund
Verifierbarhet	Boka en biljett och verifiera sedan att den skapats i administrationsgränssnittet.

Kravnummer	3
Krav	Bekräftelsemail
Beskrivning	Efter att en kund genomfört en bokning ska en bekräftelse skickas via epost till den epostadress som kunden angett i bokningen. Denna bekräftelse ska inkludera nödvändig information om bokningen, såsom: betalningsinformation, bokningsnummer, aktuella datum och tider samt en fungerande länk för avbokning.
Motivering	Det är viktigt att kunden får en bekräftelse av bokningen med aktuell information.
Behov	Standard
Prioritet	Normal
Källa	Kund
Verifierbarhet	Genomför en bokning och verifiera att ett korrekt bekräftelsemail har skickats ut till rätt epostadress.

Kravnummer	4
Krav	Möjlighet att boka studentbiljetter utöver ordinarie
Beskrivning	Det ska vara möjligt att boka specialbiljetter (så kallade studentbiljetter) utöver de ordinarie. Denna typ av biljett har ett pris som skiljer ifrån det ordinarie.
Motivering	Kårspexet vill att det ska gå att boka rabatterade biljetter
Behov	Standard
Prioritet	Normal
Källa	Kund
Verifierbarhet	Genomför en bokning som inkluderar minst en studentbiljett och verifiera sedan att dessa syns i administrationsgränssnittet.

Kravnummer	5
Krav	Utplacering av platser för bokningar
Beskrivning	I administrationsgränssnittet ska det gå att placera ut bokningar på sektions- och stolsnivå. I detta gränssnitt ska det gå att se vilka platser som är upptagna och vilka som finns tillgängliga för utplacering för den aktuella föreställningen.
Motivering	Kårspexet behöver ha möjlighet att manuellt placera ut bokningar
Behov	Standard
Prioritet	High
Källa	Kund
Verifierbarhet	TODO

Kravnummer	6
Krav	Redigering av bokningar
Beskrivning	Administratörer ska kunna redigera befintliga bokningar. Detta inkluderar att ändra betalningsstatus (hur mycket som betalats in), om biljetterna är uthämtade eller inte, etc.
Motivering	Det måste gå att uppdatera bokningsstatus
Behov	Standard
Prioritet	High
Källa	Kund
Verifierbarhet	Gå in och välj att redigera en enskild bokning, verifiera att status går att ändra och att den sparas.

Kravnummer	7
Krav	Lägga till Teater
Beskrivning	Det ska gå att bygga nya teatrar med varierande mängd platser och sektioner.
Motivering	Då kårspexet kan byta teater mellan sina omgångar
Behov	Standard
Prioritet	Normal
Källa	Kund
Verifierbarhet	Möjlighet att lägga till en ny teater utöver existerande. Det ska gå att boka en biljett på en föreställning som går på den nya teatern.

Kravnummer	8
Krav	Lägga till Omgång
Beskrivning	Man ska kunna lägga till nya omgångar i systemet från administratörs interfacet. Annars kan man inte använda systemet på tänkt sätt.
Motivering	Annars kan man inte lägga till en ny omgång
Behov	Standard
Prioritet	High
Källa	Kårspexet
Verifierbarhet	Det ska gå att från administratörens interface lägga till en ny omgång. Denna ska sedan synas i databasen.

Kravnummer	9
Krav	Redigera Omgång
Beskrivning	Man ska kunna ändra på priserna, synlighet, teater och beskrivning för en omgång.
Motivering	Om något skulle bli fel vid uppläggning av en ny omgång.
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Efter att man har lagt till en omgång i systemet ska man kunna ändra information om den

Kravnummer	10
Krav	Lägga till nya föreställningar
Beskrivning	Man ska kunna lägga till nya föreställningar i systemet från administratörs vyn.
Motivering	Varför behövs kravet? (Frivilligt fält)
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Man kan lägga till en nya föreställning via administratörs vyn.

Kravnummer	11
Krav	Redigera föreställningar
Beskrivning	Man ska kunna redigera tillagda föreställningar från administratörs vyn. Annars kan man inte uppdatera dem med ny information som är mera väsentlig.
Motivering	Utan kravet kan man inte använda systemet på tänkt sätt
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Efter att man har lagt till en föreställning ska man kunna redigera den.

Kravnummer	12
Krav	Kontantbetalning från säljarvyn
Beskrivning	En säljare ska kunna göra en kontantbetalning från säljarsystemet.
Motivering	Varför behövs kravet? (Frivilligt fält)
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Man kan välja kontant betalning i systemet och sedan utför den.

Kravnummer	13
Krav	Möjlighet att navigera i systemen
Beskrivning	Det ska finnas knappar och funktioner för att kunna navigera i de olika vyerna.
Motivering	Varför behövs kravet? (Frivilligt fält)
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Man testar att knapparna inte är trasiga länkar och att alla finns med.

Kravnummer	14
Krav	Säljare ska kunna lämna ut biljetter
Beskrivning	Säljare ska i sin vy kunna se vilka biljetter han ska lämna ut till besökaren.
Motivering	Varför behövs kravet? (Frivilligt fält)
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Man kollar att systemet ger de korrekta biljetterna som svar.

Kravnummer	15
Krav	Interaktiv översiktsbild
Beskrivning	Möjlighet för den som bokar att interagera med översiktsbilden. När man har muspekaren över en rad skall rätt sektion markeras i översiktsbilden och sektionstabellen.
Motivering	Varför behövs kravet? (Frivilligt fält)
Behov	Plus
Prioritet	Normal
Källa	Nyx
Verifierbarhet	Vid bokningssteget där man väljer sektioner skall det gå att interagera med översiktsbilden.

Kravnummer	16
Krav	Avbokning
Beskrivning	En kund skall kunna avboka sin bokning. Det genom en länk i sitt bekräftelsemail.
Motivering	Varför behövs kravet? (Frivilligt fält)
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	I bekräftelsemailet finns det en tydligt sektion med länken som gör avbokningen. Den skall fungera och ta bort bokningen från systemet om den är obetald. Förklara hur Karl ska kolla att (1) kravet finns i designen och (2) att mjukvaran implementerar kravet.

Kravnummer	17
Krav	Enklare statistik
Beskrivning	Det skall vara möjligt att från administratörens och ekonomichefens gränssnitt kunna se enklare statistik från systemet. Detta inkluderar: totalt antal utgivna biljetter per föreställning, omgång och spelår, antal utgivna biljetter som är gratis/student/ordinarie per föreställning, omgång och spelår, antal bokade biljetter per föreställning, omgång och spelår.
Motivering	Underlättar arbetet för administratör och ekonomichef inför framtida planering av nya föreställningar, omgångar och spex.
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Kontrollera att man kan se den enklare statistiken från administratörens och ekonomichefens gränssnitt.

Kravnummer	18
Krav	Omfattande statistik
Beskrivning	Det skall vara möjligt att från administratörens och ekonomichefens gränssnitt se mycket utförlig statistik från bokningssystemet. Detta inkluderar, men är ej begränsat till: enklare statistik inom vissa tidsintervall. För föreställning innebär det möjlighet att välja vilken start/sluttid bokning samt utlämning av biljetter skedde. Det skall även vara möjligt att sortera antalet bokningar/utlämningar per dag och efter eventuell rabattklass.
Motivering	För bättre förestående i bokningen vilket underlättar framtida planering och arbete av nya föreställningar, omgångar och spex.
Behov	Plus
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Vi verifierar att statistiken stämmer jämfört med testdata. Kontrollerar att det går att filtrera efter tid och eventuell rabattklass.

Kravnummer	19
Krav	Bokning Administratör
Beskrivning	Administratören skall ha möjlighet att göra bokningar via administratörsinterfacet. Administratören har full tillgång till rabattklasserna, även gratis, och kan placera ut de valda platserna direkt samt sätta status som betald. En gratisbokning är detsamma som att registrera en gratisbiljett.
Motivering	Kravet behövs för att kårspexet ska kunna ge bort gratisbiljetter och ha koll på att det är just gratisbiljetter.
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Administratören kan göra bokningar, även registrera gratisbiljetter. Osv Förklara hur Karl ska kolla att (1) kravet finns i designen och (2) att mjukvaran implementerar kravet.

Kravnummer	20
Krav	Färgkodning
Beskrivning	Den bild som ger kunden en översikt av platsfördelningen på den valda föreställningen ska ha en färgskala, som anger till vilken grad sektionerna är lediga. För att underlätta för färgblinda bör lämpliga färger väljas. Färgerna ändras dynamiskt allt eftersom fler bokningar görs.
Motivering	Ger snabb överblick för kunden i början av bokningen.
Behov	Plus
Prioritet	Normal
Källa	Nyx
Verifierbarhet	Kolla så att bilden har korrekt startfärger när det inte finns några bokningar i systemet. Kolla gradvis efter övergångar mellan färger, allt eftersom fler bokningar görs.

Kravnummer	21
Krav	Filtrerar bokningar
Beskrivning	Man ska kunna filtrera samt soterar bokningar efter väsentlig data som man själv specificerar i vyn.
Motivering	Varför behövs kravet? (Frivilligt fält)
Behov	Standard
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Man testar att systemet klarar detta.

Kravnummer	22
Krav	Kortköp
Beskrivning	Möjlighet för kund att betala med kort vid bokning.
Motivering	Varför behövs kravet? (Frivilligt fält)
Behov	Deluxe
Prioritet	Normal
Källa	Kårspexet
Verifierbarhet	Förklara hur Karl ska kolla att (1) kravet finns i designen och (2) att mjukvaran implementerar kravet.

4.1 Begränsande krav

Genererat från revision 26 av "URDSpecifika_kravKravbegränsning".

4.1.1 Prestanda

Krav	serverbelastning
Beskrivning	Systemet ska utan märkbara problem hantera minst tio typiska användare samtidigt.
Motivering	Det kommer förekomma fall då flera använder systemet samtidigt.
Behov	Standard
Prioritet	Låg
Stabilitet	Stabilt, men beror mycket på servern.
Källa	Johan Stjernberg, Kalle Arvidsson
Verifierbarhet	Skarp testning av systemet på Kårspexets server med tio eller fler användare.

Krav	svarstid
Beskrivning	Systemet får inte ta för lång tid på sig att svara på användarens anrop. Vi kan dock inte ansvara för fördröjningar i nätverket mellan systemet och användaren. Olika operationer kan ha olika långa maximala svarstider. Vid alla operationer i alla gränssnitt som enbart gäller en enstaka bokning ska systemet svara på max 1 sekund.
Motivering	Svarstiden är viktig för användarens upplevelse av systemet och vid väldigt långa svarstider försämras systemets användbarhet.
Behov	Standard
Prioritet	Låg
Stabilitet	Stabilt
Källa	Gruppen
Verifierbarhet	Systemet mäter tiden för operationen och inkluderar det i svaret (en webbsida).

4.1.2 Tillförlitlighet

Krav	driftstabilitet
Beskrivning	Medeltiden till fel i systemet som gör systemet oanvändbart ska vara mer än en vecka vid normal användning.
Motivering	Krascher av systemet gör det otillgängligt och skapar mer arbete för de som driver systemet.
Behov	Standard
Prioritet	Medel
Stabilitet	Grov uppskattning, halvstabilt.
Källa	Gruppen
Verifierbarhet	Svårt att mäta, systemet behöver köras under lång tid.

Krav	datastabilitet
Beskrivning	Systemet måste bete sig korrekt även vid en krasch, det betyder att dataförlust eller datakorruption inte får förekomma. Dataförlust sker om data som presenterats som sparade försvinner.
Motivering	Bokningar bör inte försvinna etc.
Behov	Standard
Prioritet	Medel
Stabilitet	Stabilt
Källa	Gruppen
Verifierbarhet	Vi litar på att databasen är korrekt i detta avseende. Systemet ska inte presentera information som sparad innan den sparats i databasen, inspektera koden manuellt.

4.1.3 Säkerhet

Krav	säkerhet
Beskrivning	Alla gränssnitt, förutom teaterbesökarens, ska bara kunna användas om man loggat in med lösenord.
Motivering	Bara kårspexets personal ska kunna använda dessa funktioner.
Behov	Standard
Prioritet	Låg
Stabilitet	Stabilt
Källa	Kårspexet
Verifierbarhet	Försök komma åt gränssnitten, via deras webbadress(?), se att lösenord efterfrågas, se att bara det rätta lösenordet fungerar. mer?

4.1.4 Miljö

Krav	rails
Beskrivning	Ruby on Rails är ett ramverk för utveckling av webbapplikationer. Bokningssystemet ska huvudsakligen vara byggt med detta ramverk.
Motivering	Gruppen tycker det verkar passande för projektet och vill arbeta i ramverket.
Behov	Standard
Prioritet	Hög
Stabilitet	Stabilt
Källa	Gruppen
Verifierbarhet	Undersökning av serverns konfiguration samt källkoden.

Krav	webb
Beskrivning	Kunder såväl som Kårspexets personal ska kunna använda bokningssystemet genom webbgränssnitt.
Motivering	Smidigast eftersom det innebär maximal tillgänglighet.
Behov	Standard
Prioritet	Hög
Stabilitet	Stabilt
Källa	Kårspexet
Verifierbarhet	Provkörning av systemet via webbläsare.

Krav	webbläsare
Beskrivning	Det ska gå smidigt att använda bokningssystemet i moderna versioner av dessa webbläsare: <i>Firefox</i> , <i>Internet Explorer</i> och <i>Google Chrome</i> .
Motivering	Dessa webbläsare är stora på marknaden just nu och bör stödas av vårt system.
Behov	Standard (?)
Prioritet	Låg
Stabilitet	Halvstabilt, inte alltid helt lätt att anpassa en webbsida för flera olika webbläsare och versioner.
Källa	Johan Stjernberg, Kalle Arvidsson
Verifierbarhet	Provkörning av systemet i dessa webbläsare.

4.1.5 Användbarhet

Krav	bokningstid
Beskrivning	En typisk kund ska kunna genomföra sin första bokning på mindre än fem minuter.
Motivering	Det ska vara enkelt och smidigt att använda systemet.
Behov	Standard
Prioritet	Hög
Stabilitet	Stabilt så länge inte designen ändrar sig radikalt.
Källa	Johan Stjernberg, Kalle Arvidsson
Verifierbarhet	En urvalsgrupp som inte tidigare använt systemet probokar under tidsmätning.

Krav	inlärningsstid
Beskrivning	En typisk KTH-student ska kunna sätta sig in i hur administrationsgränssnittet huvudsakligen fungerar på en dag.
Motivering	Det ska gå smidigt för Kårspexets personal att använda systemet.
Behov	Standard
Prioritet	Medel
Stabilitet	Halvstabilt
Källa	Johan Stjernberg, Kalle Arvidsson
Verifierbarhet	Undersökning av hur lång tid det tar för Kårspexets personal eller andra KTH-studenter att sätta sig in i systemet.

Krav	introduktion
Beskrivning	Vid leverans av produkt ska en introduktion till systemet ges vid ett tillfälle. Vi utlovar ingen vidare kundhjälp efter leverans.
Motivering	Det är nödvändigt att ge instruktioner till Kårspexet, dock kan inte gratis hjälp utlovas efter leverans.
Behov	Standard
Prioritet	Låg
Stabilitet	Stabilt
Källa	Johan Stjernberg, Kalle Arvidsson
Verifierbarhet	Kårspexet kan ombedas intyga att de fått instruktioner för systemet.

Krav	förbättring
Beskrivning	Kårspexets personal såväl som deras kunder ska ha ett bättre bokningssystem än det tidigare.
Motivering	Om inte vårt bokningssystem är bättre än det befintliga har vi misslyckats med vårt uppdrag.
Behov	Standard
Prioritet	Hög
Stabilitet	Stabilt
Källa	Kårspexet
Verifierbarhet	Kårspexets personal ombedes lämna en muntlig eller skriftlig jämförelse av systemen, med särskilt fokus på användbarhet och effektivitet.

4.1.6 Externa system

Krav	mysql
Beskrivning	Bokningssystemet ska använda databashanteraren MySQL.
Motivering	Gruppen vill använda MySQL och Kårspexet har samtyckt.
Behov	Standard
Prioritet	Hög
Stabilitet	Stabilt
Källa	Gruppen
Verifierbarhet	Uppvisning av databas eller kontroll av källkod.

Krav	kortbetalning
Beskrivning	Bokningssystemet ska använda sig av något externt system för kortbetalning.
Motivering	Vi kan inte ta på oss att hantera säkra kortbetalningar själva, ett externt system behövs. Genomförs i mån av tid.
Behov	Plus
Prioritet	Medel
Stabilitet	Stabilt
Källa	Gruppen
Verifierbarhet	Uppvisning av databas eller kontroll av källkod.

(Vad kallas de personer som driver/övervakar/startar om vid krasch ? Finns det sådana?)

(Ska källan vara "gruppen" eller gruppmedlemmens namn, när källan finns i gruppen?)

(Ska prioriteterna heta low, medium och high, eller ska vi ha svenska ord?)