

# A3B

Designdokument  
version 1.0

Andersson, Adam, adaan690  
Andersson, Conny, conan414  
Andersson, Martin, maran703  
Bolin, Andreas, andbo467

2012-11-09

# Innehållsförteckning

1. Inledning.....	3
2. Klassdiagram.....	4
3. Klassbeskrivning.....	5
4. Användargränssnitt.....	11

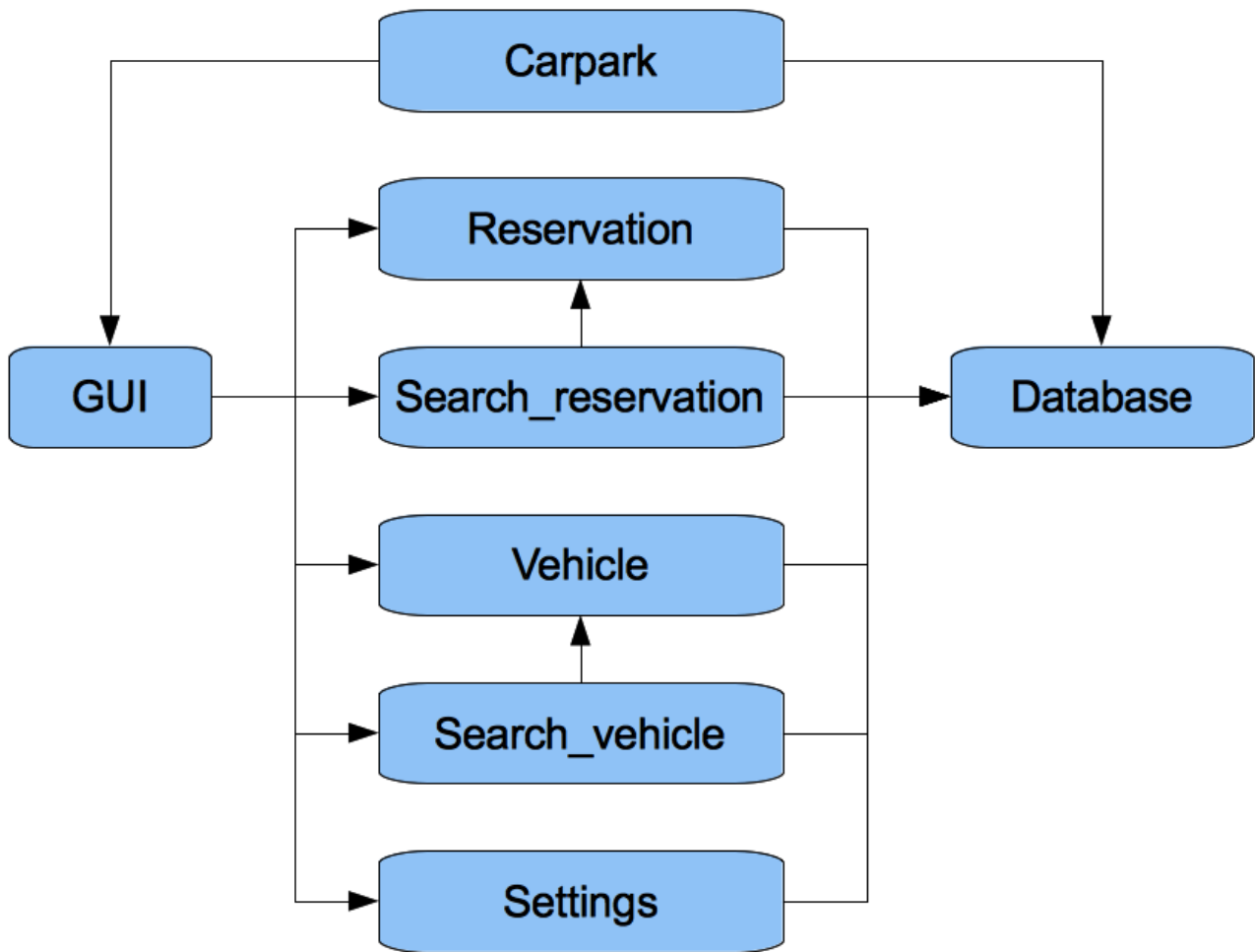
## **1. Inledning**

Detta dokument är en objektorienterad design av vårt system. Projektet utförs i kursen TDDC76 *Programmering och datastrukturer*, vid Linköpings Tekniska Högskola.

Systemet är ett bokningssystem för fordon med ett grafiskt gränssnitt mot operatören. I bokningssystemet ska man kunna boka fordon och även kunna hantera in- och utlämning av fordon. Vidare ska man även kunna söka fram bokningar och fordon. En databas kommer användas för bokningar samt en för fordon. Systemet ska täcka administrationsbehovet för en mindre uthyrningsstation.

## 2. Klassdiagram

Nedan följer ett klassdiagram för vårt system. Observera att samtliga pilar/triangular anger association samt att alla relationer är angivna med riktning.



### 3. Klassbeskrivning

Nedan följer klassformulär för våra klasser.

<b>Namn:</b>	Carpark
<b>Syfte:</b>	Övergripande klass, fungerar som en spindel i nätet.
<b>Konstruktörer:</b>	<b>Carpark()</b> <i>default konstruktor</i> <b>~Carpark()</b> <i>destruktor</i>
<b>Operationer:</b>	<b>void run()</b> <i>startar gui, öppnar databas, skapar sökobjekt</i>
<b>Datamedlemmar:</b>	-

<b>Namn:</b>	GUI
<b>Syfte:</b>	Övergripande klass, fungerar som abstraktionsbarriär mot användaren. Innehåller alla grafiska klasser.
<b>Konstruktörer:</b>	<b>GUI()</b> <i>default konstruktor</i> <b>~GUI()</b> <i>destruktor</i>
<b>Operationer:</b>	<i>Se 4. Användargränssnitt nedan för en skiss av GUI:t.</i>
<b>Datamedlemmar:</b>	<b>Reservation          current_reservation</b> <b>Search_reservation   current_reservation_result</b> <b>Search_vehicle        current_vehicle_result</b>

<b>Namn:</b>	Database
<b>Syfte:</b>	Hanterar permanenta data i systemet. Datan lagras på fil. Sqlite används som databas. Statisk klass.
<b>Konstruktörer:</b>	- Statisk, ingen konstruktor <b>~Database()</b>
<b>Operationer:</b>	public: <b>bool open(string&amp; filename)</b> Öppnar databasen i filename, alternativt skapar databas med namn filename och initerar den, returnerar true om lyckat, false annars.  <b>vector&lt;vector&lt;string&gt;&gt; reservation_search(string&amp; what, string&amp; value)</b> Gör en sökning i Reservation-tabellen i kategorin <i>what</i> efter <i>value</i> . returnerar vektor med alla rader, där varje rad består av en vektor med varje kolumn som ett element. what kan vara: res_nr, reg_nr, status,

	<p>name.</p> <p><b>vector&lt;vector&lt;string&gt;&gt; reservation_search_date(int start_date, int end_date, int start_time = 0, int end_time = 2330)</b>  Gör en sökning i Reservation-tabellen i efter bokningar mellan start_date, start_time och end_date, end_time. Returnerar vektor med alla rader, där varje rad består av en vektor med varje kolumn som ett element.</p> <p><b>vector&lt;vector&lt;string&gt;&gt; vehicle_search(string&amp; what, string&amp; value)</b>  Gör en sökning i Vehicle-tabellen i kategorin <i>what</i> efter <i>value</i>. Returnerar vektor med alla rader, där varje rad består av en vektor med varje kolumn som ett element. what kan vara: all, reg_nr, type, status, brand, model.</p> <p><b>vector&lt;vector&lt;string&gt;&gt; vehicle_search(string&amp; type, int start_date, int end_date, int start_time = 0, int end_time = 2330)</b>  Gör en sökning i Vehicle-tabellen efter ej reserverade fordon av typ <i>type</i> under start_date, start_time till end_date. end_time intervallet. Returnerar vektor med alla rader, där varje rad består av en vektor med varje kolumn som ett element.</p> <p><b>string info_search(string&amp; what, string&amp; value)</b>  Gör en sökning i Info-tabellen i kategorin <i>what</i> efter <i>value</i>. Returnerar resultatet som en sträng.</p> <p><b>void reservation_update(string&amp; res_nr, string&amp; reg_nr, int start_date, int end_date, int start_time, int end_time, string&amp; status, string&amp; name, string&amp; tel, string&amp; adress, string&amp; postal_nr, string&amp; city)</b>  Uppdaterar given bokning med de nya värdena</p> <p><b>void vehicle_update(string&amp; reg_nr, string&amp; type, string&amp; status, string&amp; brand, string&amp; model)</b>  Uppdaterar givet fordon med de nya värdena</p> <p><b>void settings_update(...)</b>  Uppdaterar Settings-tabellen.</p> <p>private:  <b>vector&lt;vector&lt;string&gt;&gt; ask(string&amp; query)</b>  Gör en förfrågan till databasen, returnerar vektor med alla rader, där varje rad består av en vektor med varje kolumn som ett element.</p> <p><b>bool exists_res(string&amp; res_nr)</b>  Returnerar sant om bokningen med res_nr redan finns i databasen</p> <p><b>bool exists_vehicle(string&amp; reg_nr)</b>  Returnerar sant om fordonet med reg_nr redan finns i databasen</p>
--	--

	<b>string int_to_string(int)</b>  <b>void close()</b> Stänger databas anslutningen
<b>Datamedlemmar:</b>	<b>sqlite3* db</b>

Databasen ska ha bestå av tabellerna: Reservations, Vehicle och Settings.

Reservations						
res_nr	reg_nr	start_date	end_date	start_time	end_time	...
12345	ABC123	20120101	20120102	1030	1030	

...	status	name	tel	address	postal_nr	city
	aktiv / kommande / avslutad	För Efter	0707123456	Storgatan 1	12345	Storstaden

Vehicle						
reg_nr	type	status	brand	model	mileage	damage
ABC123	small_car	uthyrd / tillgänglig	Ford	Ka	12324	Hö backspegel saknas, repa ovanför vä bakhjul,

Settings						
open_hour	close_hour	min_rental	...			
0800	2200	4	...			

<b>Namn:</b>	Reservation
<b>Syfte:</b>	Att skapa en ny reservation, eller möjlighet till att modifiera en redan befintlig reservation. Hjälper även till att kapsla informationen kring en bokning.
<b>Konstruktörer:</b>	<b>Reservation(datamedlemmar)</b> parametriserad konstruktor <b>Reservation(Reservation&amp;)</b> kopieringskonstruktor <b>Reservation(Reservation&amp;&amp;)</b> flyttkonstruktor <b>~Reservation()</b> destruktor

<b>Operationer:</b>	<b>void save()</b> sparar bokningen i databasen <b>void set_ "datamedlem"("värde")</b> <b>"typ" get_ "datamedlem"()</b>
<b>Datamedlemmar:</b>	<b>int res_nr</b> reservationsnummer <b>string reg_nr</b> registreringsnummer <b>string name</b> för- och efternamn <b>string tel</b> telefonnummer <b>string adress</b> <b>string postal_nr</b> <b>string city</b> <b>string status</b> aktiv / kommande / avslutad <b>int start_time</b> ex. 800 = 08:00, 1330 = 13:30 <b>int end_time</b> <b>int start_date</b> ex. 20121212 = 2012-12-12 <b>int end_date</b>

<b>Namn:</b>	Search_reservation
<b>Syfte:</b>	Att skapa en lista med befintliga reservationer baserad på olika sökparametrar.
<b>Konstruktörer:</b>	<b>Search_reservation()</b> Default konstruktör <b>~Search_reservation</b> Destruktor
<b>Operationer:</b>	<b>void res_nr(string&amp; res_nr)</b> <b>void reg_nr(string&amp; reg_nr)</b> <b>void name(string&amp; name)</b> <b>void tel(string&amp; tel)</b> <b>void status(string&amp; status)</b> <b>void start_end(int start_date,</b> <b>int end_date,</b> <b>int start_time = 0,</b> <b>int end_time = 0)</b> funktionerna ovan gör en sökning i databasen och sparar resultatet som bokningsobjekt i search_result <b>bool empty()</b> returnerar <b>true</b> om search_result är tom, <b>false</b> annars <b>void clear()</b>



	rensar search_result
<b>Datamedlemmar:</b>	<b>vector&lt;Reservation*&gt; search_result</b>

<b>Namn:</b>	Vehicle
<b>Syfte:</b>	Ska kapsla information om ett fordon och hantera ändring av denna.
<b>Konstruktörer:</b>	<b>Vehicle(string&amp; reg_nr, string&amp; type, string&amp; status, string&amp; brand, string&amp; model, int mileage, string&amp; damage)</b> parametriserad konstruktör <b>~Vehicle()</b> destruktör
<b>Operationer:</b>	<b>void set_ "datamedlem"("värde")</b>  <b>"typ" get_ "datamedlem"()</b>  <b>void save()</b> sparar/uppdaterar informationen till/i databasen.
<b>Datamedlemmar:</b>	<b>string reg_nr</b> <b>string type</b> <b>string status</b> uthyrd / tillgänglig <b>string brand</b> <b>string model</b> <b>int mileage</b> <b>string damage</b>

<b>Namn:</b>	Search_vehicle
<b>Syfte:</b>	Att skapa en lista med befintliga fordon baserad på olika sökparametrar.
<b>Konstruktörer:</b>	<b>Search_vehicle()</b> Default konstruktör <b>~Search_vehicle()</b> Destruktör
<b>Operationer:</b>	<b>void all()</b> <b>void reg_nr(string&amp; reg_nr)</b> <b>void type(string&amp; type)</b> <b>void status(string&amp; status)</b> <b>void brand(string&amp; brand)</b> <b>void brand_model(string&amp; brand, string&amp; model)</b> <b>void type_date(string&amp; type, int start_date, int end_date</b> <b>int start_time = 0, inte end_time = 2330)</b> funktionerna ovan gör en sökning i databasen och sparar resultatet som pekare till Vehicle-objekt i search_result. <b>bool empty()</b> returnerar true om search_result är tom, false annars.

	<b>void clear()</b> rensar search_result.
<b>Datamedlemmar:</b>	<b>vector&lt;Vehicle*&gt; search_result</b>

<b>Namn:</b>	Settings
<b>Syfte:</b>	Att hålla reda på alla inställningar som användaren kan göra i programmet.
<b>Konstruktörer:</b>	<b>Settings()</b> default konstruktor <b>~Settings()</b> destruktor
<b>Operationer:</b>	<b>void save()</b> sparar/uppdaterar informationen till/i databasen. <b>void set_ "datamedlem"("värde")</b> <b>"typ" get_ "datamedlem"()</b>
<b>Datamedlemmar:</b>	<b>int open_hour</b> Öppnar <b>int close_hour</b> Stänger <b>int min_rental</b> Minsta uthyrningsperiod

#### 4. Användargränssnitt

Nedan följer en övergripande skiss över hur vi tänkt oss att bokningssidan för systemet ska se ut. Därefter följer hur söksidan är tänkt att se ut.

Arkiv	Inställningar	Om
-------	---------------	----

**Bokning****Sök****Lämna ut****Återlämna****Fordon**

Startdatum: År v Månad v Dag v Tid: Från 00:00 v

Slutdatum: År v Månad v Dag v Till 00:00 v

Sök

Liten bil

Mellan bil

Stor bil

Liten Lastbil

Stor Lastbil

☐ Mellan bil - Reg nr: 123 ABC Tid: 00:00-00:00

☒ Mellan bil - Reg nr: 456 DEF Tid: 00:00-00:00

☐ Mellan bil - Reg nr: 789 GHI Tid: 00:00-00:00

-----

Liknande förslag:

☐ Stor bil - Reg nr: 123 EFG Tid: 00:00-00:00

Information

Reg nr: 456 DEF

Fordonstyp: Bil

Storlek: Mellan

Märke: Mercedes

Längd: X,XX m

Bredd: X,XX m

Höjd: X,XX m

Boka >

Överst i denna flik väljs start- och slut- datum och tid. Väljs gör de med rullistor. Under detta väljs kategori av fordon vilket listar tillgängliga fordon vid given tid efter att man tryckt på knappen "Sök bokning". I listan visas fordonstyp, reg. nr., samt start- och slutdatum. Om bokningen är tillgänglig aktiveras en knapp "Boka" som tar en vidare till ett formulär där kunduppgifter matas in. Därefter ombeds man bekräfta bokningen.

Arkiv	Inställningar	Om
-------	---------------	----

Bokning	Sök	Lämna ut	Återlämna	Fordon
---------	-----	----------	-----------	--------

Bokn nr

Reg nr

Namn

Pers nr

Tlfn

Datum

☒ Avslutade
 ☒ Aktiva
 ☐ Kommande

Resultat för reg nr ABC 123

- ☐ Avslutad bokning: 120819 - 120820
- ☒ Avslutad bokning: 120827 - 120901
- ☐ Avslutad bokning: 121010 - 121012
- ☐ Aktiv bokning: 121108 - 121111

Överst visas en sökruta där operatören kan mata in data. Till vänster i fliken finns en undermeny där man väljer hur man vill söka på en bokning. T.ex. via bokningsnummer eller personnummer. När man trycker "Sök" listas de bokningar som överensstämmer och relevant information om dessa. Underst finns knapparna "Visa", "Ändra" och "Ta bort". "Visa" ger all info om vald bokning. "Ändra" skickar vidare en till bokningssidan med aktuella värden inmatade om bokningen ej är avslutad. "Ta bort" ber om en bekräftelse för att ta bort bokningen.