

# SWEN 262

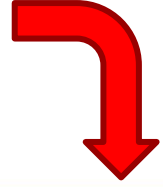
Engineering of  
Software Subsystems

*Noun & Verb Analysis*

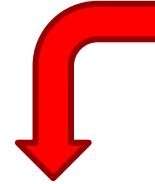
# Nouns & Verbs

- Breaking a large problem down into a class structure is referred to as *class decomposition*.
- Analyzing the nouns and verbs in the problem statement can be the first step in creating a class decomposition.
- Nouns may represent:
  - *Classes*
  - *Specializations (i.e. subclasses)*
  - *Attributes (i.e. fields)*
  - *Data elements*
- Verbs may represent:
  - *Services (i.e. methods) provided by a class*
  - *Services used by a class*

You are designing a restaurant reservation system. Each restaurant has information about its location, type of food, hours of operation, and reservations. The restaurants branch out into regular, catering, and take-out where making a reservation means a different thing for each category of restaurant. Diners will be using the system either when logged into their account or as a guest. Diners, either as guests or logged in users, can make reservations. Reservations will be linked to a user's account if they are logged in. Logged in users can also view, modify, or delete reservations in their account. A user's account will have a name, rewards points, and their upcoming and past reservations. Restaurant users will have an account which can view, modify, or delete any reservation made for their restaurant. The restaurant user can also make a reservation when diners arrive. All users will authenticate using a password or Google login.



Nouns	Verbs
restaurant [regular, catering, take-out] (location, type of food, hours of operation, reservations)	
reservation	
system	
information	
location	
hours of operation	
User [logged in, guest] (name, rewards points, password, Google login, reservations)	complete (reservation) saves (reservation) views/modifies/deletes (reservation) authenticate (password) authenticate (Google login)



Classes	Subclasses	Attributes	Responsibilities
Restaurant	Regular Catering Take-Out	Location Type of food Hours of operation Reservations	
Reservation			Make Complete Save View Modify Delete
User	Logged-In Guest	Name Rewards points Password Google login Reservations	
Location			
Hours of Operation			

# Intentional Analysis

- An intentional analysis goes beyond simply listing the nouns and verbs and provides more guidance.
- Using structure and annotating the nouns and verbs list moves you toward a class decomposition.
- This helps yield a class structure that is isomorphic to the world view of the problem domain.
- It also helps identify the location for services/responsibilities.

You are designing a restaurant reservation system. Each restaurant has information about its location, type of food, hours of operation, and reservations. The restaurants are broken up into regular, catering, and take-out where making a reservation means a different thing for each category of restaurant. Diners will be able to use the system either when logged into their account or as a guest. Diners, either as guests or logged in users, can make reservations. Reservations will be saved to a user's account if they are logged in. Logged in users can also view, modify, or delete reservations in their account. A user's account will store a name, rewards points, and their upcoming and past reservations. Restaurant users will have an account which can view, modify, or delete any reservation made for their restaurant. The restaurant user can also complete a reservation when diners arrive. All users will authenticate using a password or Google login.

We **do** typically begin by highlighting the nouns and verbs in the original problem statement, but this is just the first step.

Let's take a look at a detailed example.

# A Restaurant Reservation System

You are designing a restaurant reservation system. Each restaurant has information about its location, type of food, hours of operation, and reservations. The restaurants are broken up into regular, catering, and take-out where making a reservation means a different thing for each category of restaurant. Diners will be able to use the system either when logged into their account or as a guest. Diners, either as guests or logged in users, can make reservations. Reservations will be saved to a user's account if they are logged in. Logged in users can also view, modify, or delete reservations in their account. A user's account will store a name, rewards points, and their upcoming and past reservations. Restaurant users will have an account which can view, modify, or delete any reservation made for their restaurant. The restaurant user can also complete a reservation when diners arrive. All users will authenticate using a password or Google login.

# Step 1 – Identify the Nouns

Some nouns may obviously not make it into the design, but be thorough.

You do not need to highlight the same (or similar) nouns more than once.

You are designing a restaurant reservation system. Each restaurant has information about its location, type of food, hours of operation, and reservations. The restaurants are broken up into regular, catering, and take-out where making a reservation means a different thing for each category of restaurant. Diners will be able to use the system either when logged into their account or as a guest. Diners, either as guests or logged in users, can make reservations. Reservations will be saved to a user's account if they are logged in. Logged in users can also view, modify, or delete reservations in their account. A user's account will store a name, rewards points, and their upcoming and past reservations. Restaurant users will have an account which can view, modify, or delete any reservation made for their restaurant. The restaurant user can also complete a reservation when diners arrive. All users will authenticate using a password or Google login.

# Initial Noun List

- List all of the nouns in the left column of a two-column table.
  - *Put each noun in its own row in the table.*

Nouns	Verbs
You	
restaurant	
reservation	
system	
information	
location	
type of food	
hours of operation	
regular	
catering	
take-out	
thing	
category of restaurant	
diners	
account	
guest	
logged in user	
name	
rewards points	
password	
Google login	

# Step 2 - Identify Specializations

- Identify the nouns that might be specializations of other nouns.
  - Put the specializing word in square brackets ([ ]) next to the noun it is specializing.
  - If the noun has no other use than its specialization, remove its row from the table.

Nouns	Verbs
You	
restaurant [regular, catering, take-out]	Here, specializations are shown in red just to make them more visually distinctive.
reservation	
system	
information	
location	
type of food	
hours of operation	
regular	In <b>your</b> assignments, <b>do not</b> remove the rows, but use shading to show that they are eliminated.
catering	
take-out	
thing	
category of restaurant	
diners	
account	
guest	
User [logged in, guest]	You may also combine or rearrange nouns to create specializations that make sense.
name	
rewards points	
password	
Google login	

# Step 3 - Identify Attributes

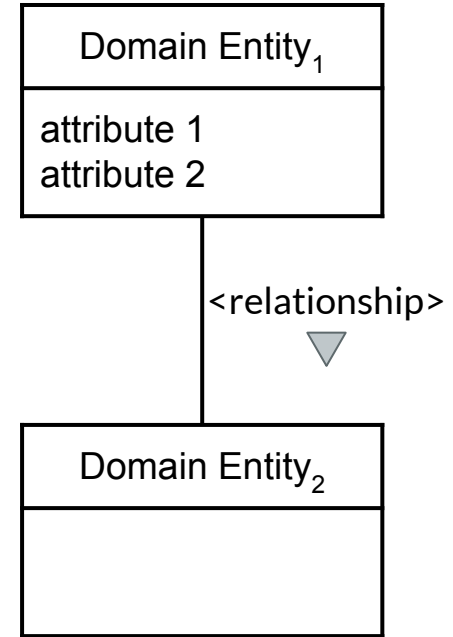
- Identify nouns that may be:
  - *Attributes of other nouns*
  - *Properties of other nouns*
  - *Information/data owned by other nouns.*
- Connect the nouns together.
  - *Put the associated noun in parentheses (()) next to the noun it is associated with.*
  - *Remove the associated nouns row only if it is a primitive data type. Err in the direction of keeping nouns in the list.*

Nouns	Verbs
You	
restaurant [regular, catering, take-out] (location, type of food, hours of operation, reservations)	Here again color is used just to make the associations visually distinctive.
reservation	
system	
information	
location	
type of food	Again, in <b>your</b> assignments, you should <b>not</b> delete the rows, but use shading.
hours of operation	
thing	
diners	
account	
User [logged in, guest] (name, rewards points, password, Google login, reservations)	
name	
rewards points	
password	
Google login	



# Domain Analysis

- Now that we have completed a thorough noun analysis, it is time to pivot to creating a **domain model**.
- A domain model uses a simplified UML class diagram notation and should show the following information:
  - Domain entities
  - Attributes
  - Relationships
- The domain model should only use language from the original domain, i.e. the nouns from your analysis.



The relationship should complete a phrase when read *Domain Entity<sub>1</sub>* *<relationship>* *Domain Entity<sub>2</sub>*.

# Domain Analysis

Restaurant
location type of food hours of operation
Regular
Catering
Take-Out

Reservation
User
name rewards points password Google login
Guest
Logged-In

Nouns
You
restaurant [regular, catering, take-out] (location, type of food, hours of operation, reservations)
reservation
system
information
location
hours of operation
thing
diners
account
User [logged in, guest] (name, rewards points, password, Google login, reservations)

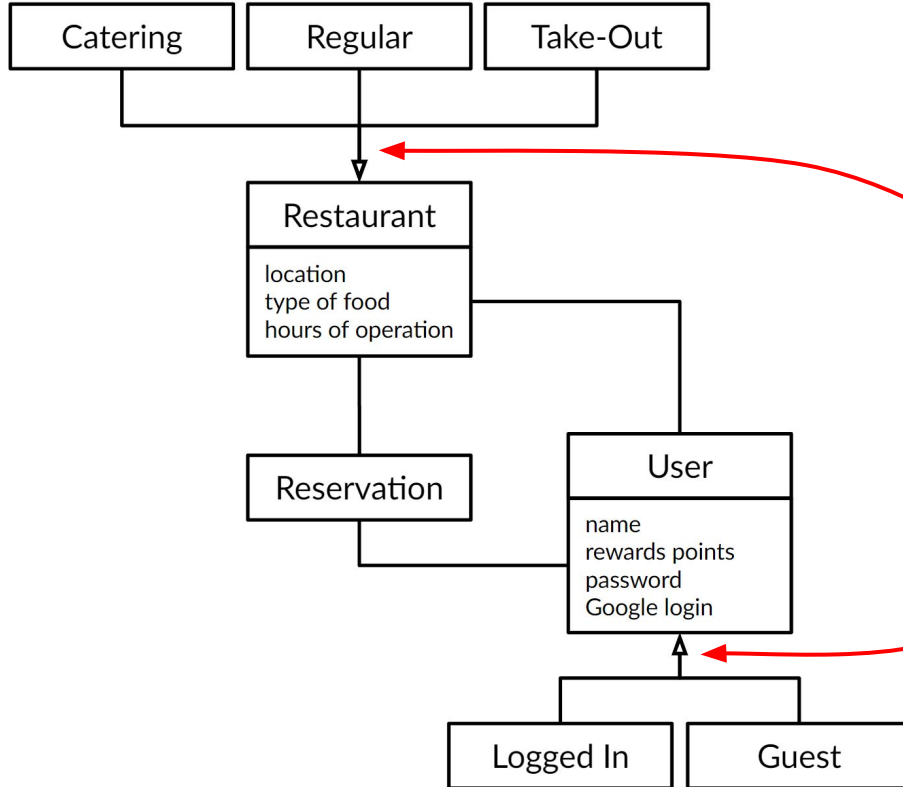
Each noun in the analysis is a potential entity in the domain mode.

But we can make some educated guesses about nouns that probably won't be in the model.

For example, nouns that seem unimportant (e.g you, thing) or redundant (e.g. account, diners).

As well as nouns that appear as the attribute of another noun and don't seem important enough to be a separate entity.

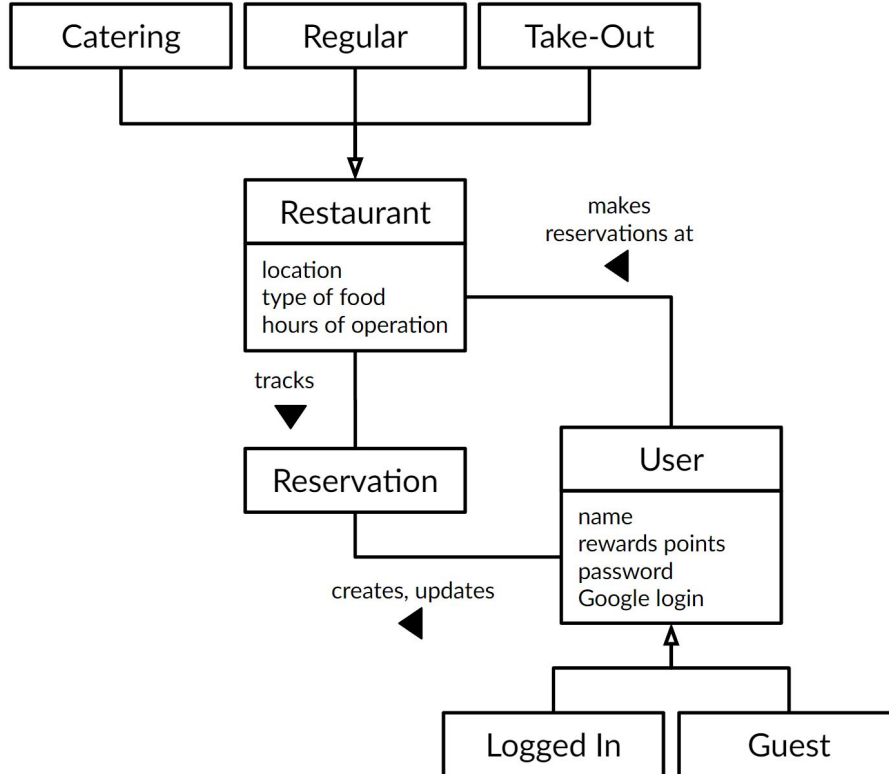
# Domain Analysis



The next step is to connect the entities that are *related* to each other in some way.

If the relationship is *inheritance*, then we use the standard UML notation for that.

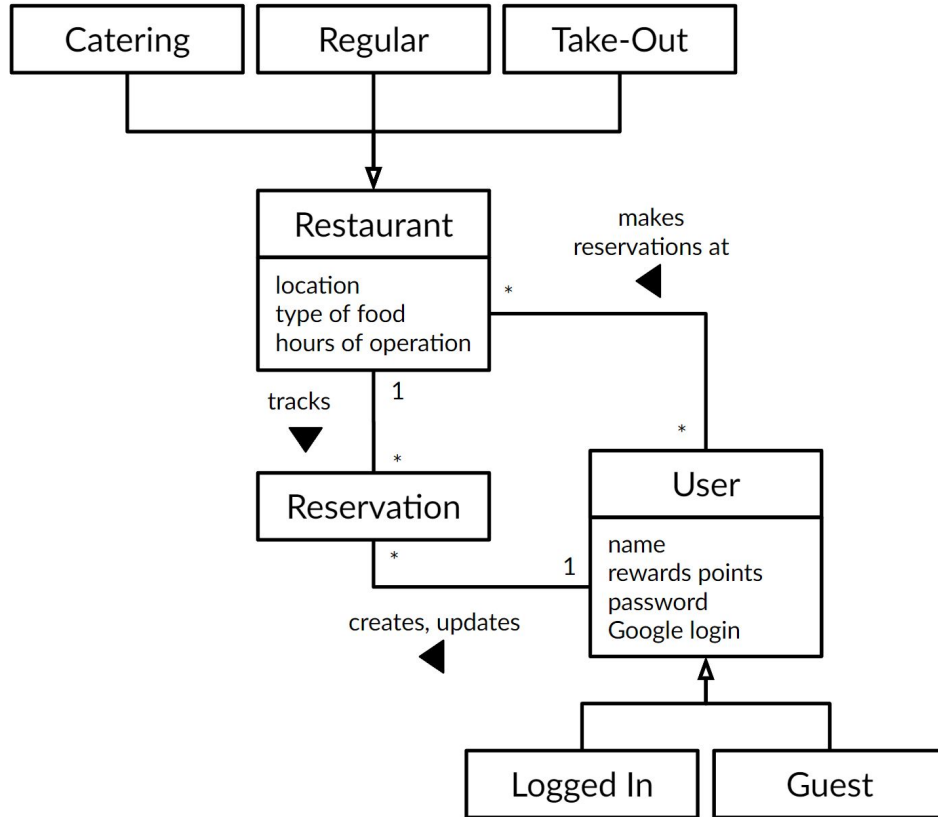
# Domain Analysis



Every relationship should be labeled using a **descriptive phrase** directly from the problem statement.

**Arrows** should be used to indicate the **direction** of the relationship.

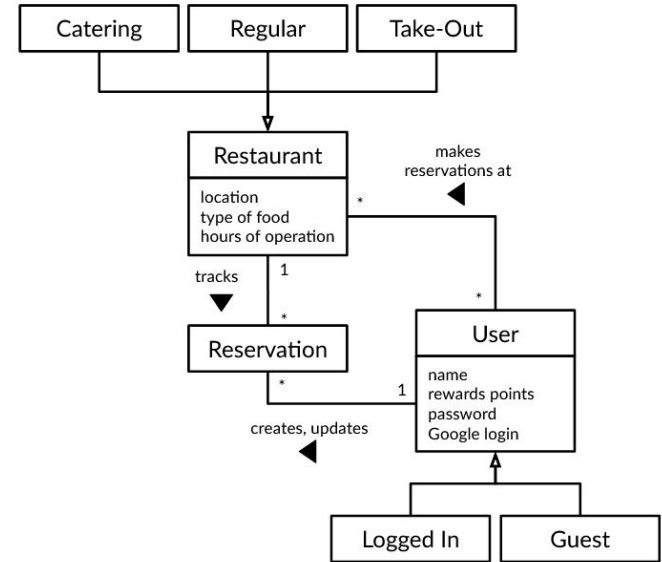
# Domain Analysis



Finally, be sure to indicate the **multiplicity** for any relationships that are not 1-to-1.

# Interlude – Domain Analysis

- The purpose of a domain analysis is to start a conversation with the customer.
  - *The model is presented to the customer along with the question: did I get this right?*
  - *The customer will provide feedback and request changes to the model.*
  - *The purpose of the conversation is to establish a shared understanding of the customer's vision **before** beginning a detailed design of the system.*
  - *As a result of the conversation with your customer, you may revise the model before continuing.*



For the purposes of this exercise, we will assume that the domain model presented accurately captured the customer's vision of the product.

# Next – Identify the Verbs

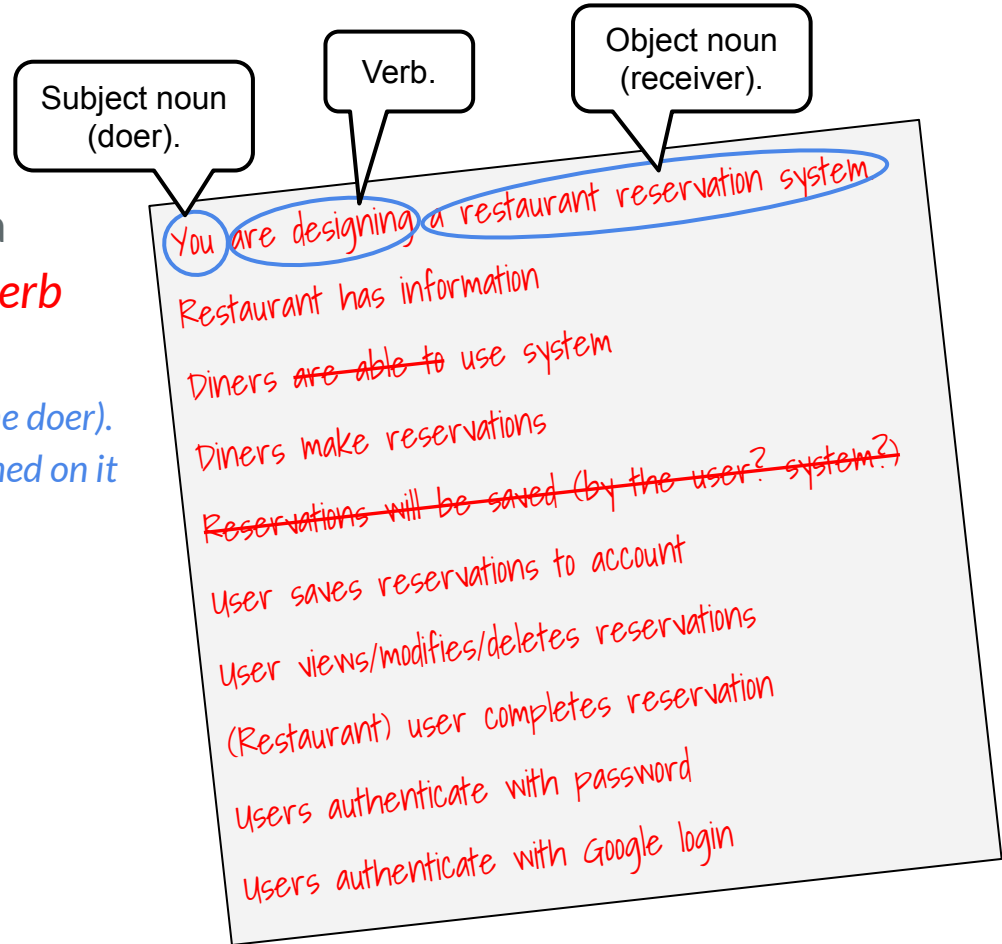
Just as with nouns, some verbs may obviously not make it into the design, but be thorough.

You may want to highlight the same verb more than once if it appears in different contexts.

You are designing a restaurant reservation system. Each restaurant has information about its location, type of food, hours of operation, and reservations. The restaurants are broken up into regular, catering, and take-out where making a reservation means a different thing for each category of restaurant. Diners will be able to use the system either when logged into their account or as a guest. Diners, either as guests or logged in users, can make reservations. Reservations will be saved to a user's account if they are logged in. Logged in users can also view, modify, or delete reservations in their account. A user's account will store a name, rewards points, and their upcoming and past reservations. Restaurant users will have an account which can view, modify, or delete any reservation made for their restaurant. The restaurant user can also complete a reservation when diners arrive. All users will authenticate using a password or Google login.

# Expressing Verbs

- For each verb, express its use in a phrase of the form **subject-noun verb object-noun**.
  - The subject noun invokes the action (the doer).
  - The object noun has the action performed on it (the receiver).
- Use the active voice for the verb.
  - If it is in the passive voice, rephrase it.





# Connect Verbs to Nouns

- Connect each verb to the nouns in the phrase expressing the verb's use.
  - *Put each verb in the right column and the same row as the subject noun (the doer).*
  - *Put the object noun (the receiver) in parentheses (()) next to the verb.*

Nouns	Verbs
You	designing (restaurant reservation system)
restaurant [regular, catering, take-out] (location, type of food, hours of operation, reservations)	has (information)
reservation	<div>Here again color is used just to make the associations visually distinctive.</div>
system	
information	
location	
hours of operation	
thing	
diners	use (system) make (reservation)
account	
User [logged in, guest] (name, rewards points, password, Google login, reservations)	complete (reservation) saves (reservation) views/modifies/deletes (reservation) authenticate (password) authenticate (Google login)

# Pruning

- If a noun has no attributes, no specializations, and does not invoke or receive any verbs, consider eliminating it from the table.
  - *Is there a verb missing that this noun invokes or expresses an action performed on this noun?*
  - *Is this noun really outside of the system boundary?*
- Are some nouns synonyms?
- Are some verbs of little relevance?

Nouns	Verbs
You	designing (restaurant reservation system)
restaurant [regular, catering, take-out] (location, type of food, hours of operation, reservations)	has (information)
reservation	<div>Again, in <b>your</b> assignments, you should <b>not</b> delete the rows, but use shading.</div>
system	
information	
location	
hours of operation	
thing	
diners	use (system) make (reservation)
account	
User [logged in, guest] (name, rewards points, password, Google login, reservations)	complete (reservation) saves (reservation) views/modifies/deletes (reservation) authenticate (password) authenticate (Google login)

# First-Cut Class Decomposition

- Nouns in a table row (row-nouns) are potential classes.
- Nouns in the left in parentheses are potential attributes (fields) of the row-noun.
- Nouns on the left in brackets are potential subclasses with the row-noun as the superclass.
- Verbs on the right are responsibilities potentially defined as methods in the object-nouns (in parentheses).
  - *A noun that is “responsible” for a verb action contains the code that implements that action.*

The “row-nouns” are potential classes.

Nouns in square brackets are potential subclasses.

Nouns	Verbs
restaurant [regular, catering, take-out] (location, type of food, hours of operation, reservations)	
reservation	
system	
information	
location	
hours of operation	
User [logged in, guest] (name, rewards points, password, Google login, reservations)	complete (reservation) save (reservation) views/modifies/deletes (reservation) authenticate (password) authenticate (Google login)

Nouns on the left in parentheses are potential fields.

Nouns on the right in parentheses potentially implement the associated verbs.

# First-Cut Class Decomposition Table

Classes	Subclasses	Attributes	Responsibilities
Restaurant	Regular Catering Take-Out	Location Type of food Hours of operation Reservations	
Reservation			Make Complete Save View Modify Delete
User	Logged-In Guest	Name Rewards points Password Google login Reservations	
Location			
Hours of Operation			

# CRC Cards

- The first-cut class decomposition includes high level information about classes/subclasses, attributes, and responsibilities.
- The next step is to spend time articulating the precise role and responsibilities of each class in the system.
- One common mechanism for doing this is a **Class-Responsibilities-Collaborators (CRC) Card**.

<b>Class:</b> <i>ClassName</i>	
<b>Responsibilities:</b> <i>A description of the class's responsibility within the context of the system. This description should be <u>at least</u> 2-3 sentences long.</i>	
<b>Collaborators</b> <i>(do not write anything here)</i>	
<b>Uses:</b> <i>The list of classes that this class uses (depends on).</i>	<b>Used By:</b> <i>The list of classes that use (depend on) this class.</i>
<b>Author:</b> <i>The team member(s) that wrote this class.</i>	

# CRC Example

<b>Class:</b> Restaurant	
<b>Responsibilities:</b> The overall responsibility of this class is to represent one of the restaurants at which users of the system may make reservations. The class includes information about the restaurant's location, the type of food served, hours of operation, and maintains a list of upcoming and past reservations.	
<b>Collaborators</b>	
<b>Uses:</b> Reservation, Hours of Operation, Location	<b>Used By:</b> RegularRestaurant, CateringRestaurant, TakeOutRestauarant, User
<b>Author:</b> J. Smith	