CS310 Assignment #1 - Requirements Specification

**Deadline:** 09/21, Wednesday, 11:59:59pm

**Late policy:** Late submissions lose 1% of the total available points for each hour the assignment is late. Assignments submitted more than 48 hours after the original deadline will not be accepted.

**Submission:** All submissions will be via Blackboard. Each team only needs to submit one PDF file under “Assignment -> Assignment 1”.

**Team size:** 5-6 students

**Background**

**Sharing Economy**

Life is all about sharing. There are always situations where we have extra resources that we don’t need, or we lack resources that other people may have. Nowadays, the pervasiveness of connectivity provides the opportunity for people to share their resources more easily. Thus, a platform for people to share their resources directly without having to go through troublesome intermediate will largely benefit the society. In fact, there are a lot of emerging startups share the same goal [1]. This kind of business model is called “Sharing Economy”, as illustrated in Fig.1. In a nutshell, there are three parties in a sharing economy, i.e., owner, seeker, and platform provider. Owners have extra resources that they can share, seekers look for resources they are interested in, and platform providers supply the (usually, software) platform for owners and seekers to interact directly.

**PRSP**

Your course project is to develop a Peer-to-Peer Resource Sharing Platform (PRSP). In other words, you are a PRSP provider. For the purpose of this assignment, both owners and seekers are called “peers”, and they interact with other peers directly without the need for central coordination. To be more specific, a PRSP has the following three key characteristics [2].

1. **Easy Entrance**
   - any individual can be owner/seeker (i.e., they do not need to be part of an organization)

2. **Peer-to-Peer Structure**
   - owners and seekers interact directly
   - the platform provider does not own any resources, other than the platform itself

3. **Public Sharing**
   - resources are shared to the public (rather than being kept within a private group)

You can find a list of examples of PRSPs below. Some counterexamples are Amazon (owners, i.e., sellers, are Amazon itself and Amazon’s partners), Priceline (it acts as the intermediary in
all customer transactions), **Budget** (the company owns the cars), and many, many others in the “traditional” economy.

**Sharing Economy**

![Sharing Economy Diagram](http://bmtoolbox.net/patterns/sharing-economy/)

Fig. 1. [http://bmtoolbox.net/patterns/sharing-economy/](http://bmtoolbox.net/patterns/sharing-economy/)

A Closer Look at a PRSP with Which You May Be Familiar

**Platform Provider: Airbnb**

**Resource: Apartment**

**Owner: Host**

**Seeker: Traveler**

Characteristics (Compared to, e.g., Hilton Hotels):

<table>
<thead>
<tr>
<th></th>
<th>Airbnb</th>
<th>Hilton Hotels</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Easy Entrance</strong></td>
<td>Everyone can be a host</td>
<td>The Hilton corporation is the host</td>
</tr>
<tr>
<td><strong>P2P Structure</strong></td>
<td>Host and traveler interact directly; Airbnb doesn’t own any apartments</td>
<td>Traveler interacts with customer service, not the actual owner of the room; Hilton owns the rooms</td>
</tr>
<tr>
<td><strong>Public Sharing</strong></td>
<td>Everyone can search for any published apartments</td>
<td>Some rooms are reserved for a specific group (e.g. Presidential Suite)</td>
</tr>
</tbody>
</table>
The following figures will help us zoom in to see how Airbnb works, from different perspectives.

**How Airbnb works**

1. The user searches on Airbnb, where people rent rooms or homes by days.
2. Previous users rate the host and comment on conditions of the place, problems, etc.
3. Once the user finds a suitable place that is available, it can be booked through Airbnb.
4. The payment is done through Airbnb, which gets a percentage of the price (around 6.13% from clients, 3% from hosts).

**Fig. 2. Airbnb in a nutshell (traveler perspective)**

**Fig. 3. Airbnb whole process (host perspective)**

**Fig. 4. Airbnb payment process**

Additional Examples of PRSPs with Which You May Be Familiar

- **Transportation:** e.g., Uber, Lyft
- **Time & Knowledge:** e.g., Got it
- **Money:** e.g., Zopa
People: e.g., Match.com
Pets: e.g., DogVacay

Project Description

Your team will work on building ParkHere, a P2P Resource Sharing Platform (PRSP), where the resources that will be shared are private parking spaces throughout the city. A city such as Los Angeles has a number of areas where the parking is scarce and/or can be really expensive. Examples are the areas around USC and UCLA, Downtown LA, many areas close to the beaches, big tourist attractions such as the Hollywood Walk of Fame, event venues such as the Hollywood Bowl, etc. People who live close to those areas and/or have previously purchased parking spaces may be willing to share the available parking spaces for a price. It is to be expected that the price of parking obtained through ParkHere, while dependent on the availability, time, duration, etc., would be lower than the parking available through existing options, such as “fixed price” public parking lots. Furthermore, the added flexibility of knowing the exact location of parking ahead of time may be a big attraction to users.

In this assignment, you will focus on the requirements for the ParkHere system, and will deliver a requirements specification using one or more of the notations discussed in class on August 31, and in Chapters 4 and 5 of the textbook. Please note that Airbnb and other examples provided above are actual products and, as such, they are more complicated than a course project. They are meant to help you form your idea rather than to indicate or describe the scope of your project.

Your team has a customer - Yixue Zhao. Yixue has several requirements for the system that she wants you to meet. You will work with your customer closely throughout the semester and your goal is to deliver a project that satisfies your customer. For this assignment, you will have one class session, on September 7, to obtain the initial user requirements and to elicit the necessary information to convert them into system requirements (recall the lecture of August 31 and Chapter 4). You may subsequently use email and/or visit Yixue’s regular office hours to clarify and refine those requirements further, as necessary. You may even schedule a separate meeting with Yixue if you can demonstrate that there is a special need or circumstance; Yixue and, if necessary, I will determine whether there is an actual need and justification for such a meeting (e.g., just because you got a late start on the assignment would not be a satisfactory reason). Finally, you will be able to get last-minute clarifications and any other feedback in class, on September 19.

Please note that

1. your customer may not have the complete technical and/or domain knowledge, and thus she may be unclear in her requirements;
2. your customer may change her requirements along the way; customers change their minds and sometimes are guided by the IKIWISI (“I'll know it when I see it”) principle;
3. your customer may have conflicting requirements; and
4. your customer may have unreasonably hard-to-implement requirements. In this particular assignment, your job is to communicate with your customer and elicit the requirements and complete the specification. Please feel free to negotiate the requirements with your customer, but be aware of your competitors (other teams) - if you cut too many corners to be able to complete the project, your customer may “purchase” another team’s solution.

Requirements

Your goal is to build ParkHere, a PRSP for private parking space sharing. The owner of a single parking space or an entire parking lot can publish the location of idle parking spaces they are willing to share to the public via your ParkHere platform. A parking-space seeker can search the area they are interested in on ParkHere. The sharing of the parking space is temporary, i.e., the parking space will only be available to the seeker for a specific time span. Your ParkHere platform should support scenarios such as the one described below. The platform should be scalable, energy-efficient, secure, fast, user-friendly, and fault tolerant.

Sample Scenario: Duc is a student who lives near USC. He has an assigned parking space at his apartment complex. Arman is a student who doesn’t live close to USC and doesn’t want to purchase a USC parking permit. This summer, Duc was doing an out-of-town internship, so he posted his parking space availability on ParkHere. Arman was taking summer classes at USC and he found Duc’s post on ParkHere, so he paid Duc and parked in Duc’s parking space on the days his classes met. This semester, Duc moved to a bigger apartment in the same complex and got an additional parking space. He also realized that parking is a big problem on game days, so he raised the price of his two parking spaces on game days.

Design and Implementation Constraints

There are only two general design and implementation constraints. Other similar constraints may emerge as you refine the requirements and elicit new ones from the customer.

1. Your front end must be a native Android App (not a web application that works inside a browser on Android).
2. Your solution must have a back end - any back-end language is acceptable, e.g. Java, PHP, Node.js, etc.

Deliverable

Your team will deliver a requirements specification document. The document should be as complete, precise, consistent, and unambiguous as possible (subject to the typical limitations, as discussed in class on August 31, and in Chapter 4 of the textbook). As a baseline, you should use a combination of plain English, with numbered paragraphs indicating different requirements and their sub-requirements (as discussed in class on August 31 and in Chapter 4), and at least one type of diagram (as discussed in class on August 31 and in Chapter 5). You are likely to find that diagrams are a convenient way of capturing requirements information and that you can use more than one type of diagram. While you are not required to use
formal/mathematical notations, correct use of them can be an effective way of conveying requirements and will be considered for extra-credit in this assignment. Finally, you are supposed to come up with at least five scenarios/user stories in the description of your system (as discussed in class on August 31 and in Chapter 4). They may be similar to the scenario described above.

Your requirements document must have the following sections. These were discussed on slides 60-61 of the August 31 lecture and in Chapter 4 of the textbook.

1. Project Title and Authors
   a. Your assigned team number
   b. Optionally a name you select for your team
   c. A list of all team members (names and USC ID numbers)
2. Preface
3. Introduction
4. Glossary
5. User Requirements Definition
   a. Scenarios/user stories
6. System Requirements Specification (all of the below)
   a. Functional Requirements
   b. Non-functional Requirements
   c. UI Appearance Requirements
7. System Models (one or more of the below)
   ○ Context Diagrams
   ○ Process Diagrams
   ○ Class Hierarchy Diagrams
   ○ Aggregation Diagrams
   ○ Use Case Diagrams
   ○ Sequence Diagrams
   ○ State-Machine Diagrams
   ○ etc.

References