



SW Engineering CSC648/848 Fall 2018

## **Milestone 2**

Team 15 (global)

Jonas Kühle, Eric Groom, Mariko Sampaga, Nay Lin Min

[jkuehle@mail.sfsu.edu](mailto:jkuehle@mail.sfsu.edu)

Oct 18 2018

Revision History Table		
Version 0.1	Submitted for Review	Oct 18 2018
Version 0.2	Revised	Oct 25 2018

<b>1. Data Definitions V2</b>	<b>3</b>
<b>2. Functional Requirements V2</b>	<b>5</b>
<b>3. UI Mockups/Storyboard (High Level)</b>	<b>6</b>
3.1 Homepage	6
3.2 Post Item	7
3.3 Registration	9
3.4 Contact Seller	10
3.5 Moderator	11
3.6 Dashboard	12
<b>4. High Level Architecture, Database Organization</b>	<b>13</b>
<b>5. High-level UML diagrams</b>	<b>18</b>
5.1 Class Diagram	18
5.2 Component Diagram	19
5.3 Deployment Diagram	20
<b>6. Identify actual key risks for your project at this time</b>	<b>21</b>

# 1. Data Definitions V2

Item:

An Item is the thing which shall be sold. It consists of an image, title, category, description and a price. Each item must be approved by a moderator.

- Title: String
  - The title of the item. Purpose is to allow buyers to browse through site using the titles of the items. Used in % search.
- Description: String
  - Description of the item. Purpose is for seller to provide more information regarding the item.
- Category: Enum
  - Category of the item. Purpose is to organize the items under a category. Used in browsing through the site without searching.
- Price: Decimal
  - Price of the item. Purpose is for buyers to have an idea of how much the item is. Used in sorting by price.
- Seller: User
  - Seller of the item. Purpose is to tie the item to a seller. Used in contacting between buyer and seller. Also, used by moderator in potential blacklisting of a user.
- Image(s): [Image], png, path to image in file system, between 1 and 5 inclusive per item.
  - Image of the item. Purpose is to have a visual representation of the item.
- Status: Enum
  - Status of the item. Purpose is to allow seller to see the status of their item. Used by moderator to approve/deny items.
  - Approve, Deny, Pending, Sold

User: A user is a person who buys, sells or searches items on our website. They are divided into 3 groups: Unregistered, registered and moderators. Unregistered users won't be in the database so this is only for registered users.

- First Name(s): String
- Last Name(s): String
- Email: String
- HashedPassword: String
- ItemsForSale: [Item]
- WishlistItems: [Item]
- IsModerator: Bool
- IsBlacklisted: Bool

Message: A text message from the buyer to the seller.

- Recipient: User
- Sender: User
- Date: DateTime
- Item: Item
- Content: String || Location

Location: Predefined spots for meetups. There will be a images of a campus map with a pin marking the meetup location.

- Location: String
- Map: Image

Category: Category for the item to be placed under. List of Categories (Electronics, Books, Furnitures, Clothing, Misc.). Attribute "CourseNumber" is only *true* is it is a book, other is *null*.

- Title: String
- isBook: Bool
- CourseNumber: String

Image: An image of the item for sale. There must be at at least 1 and max. 5 images per item.

- ImagePath: String
- ThumbnailPath: String

## 2. Functional Requirements V2

*Moderators have all rights of Registered Users and Registered Users have all rights of Unregistered Users.*

Priority 1:

- Unregistered Users
  1. Shall be allowed to browse posted items
  2. Shall be allowed to go through with the posting process but not complete it without registering (lazy registration)
  3. Shall not be able to complete posting process to sell an item until they sign up.
  4. Shall be able to filter items by category
  5. Shall be allowed to search in text bar
  18. Shall be able to sort items by price
- Registered Users
  7. Shall have access to complete the posting process to sell one or more items.
  8. Shall have access to the seller's dashboard where they can see the status of their item(s) and received messages.
  9. Shall be able to be contacted by seller(s) by providing their phone number.
  13. Shall be able to message each other through intrasite messaging.
- Moderators
  11. Shall have access to moderator dashboard where they can approve/deny items being posted.
  12. Shall be able to blacklist users.

Priority 2:

- Unregistered Users
  16. Shall be able to search for books by SFSU course number.
- Registered Users

*None*
- Moderators

*None*

Priority 3:

- Unregistered Users
  19. Shall be able to filter by color
- Registered Users
  14. While messaging about an item users shall be able to specify a meetup location from a predetermined list of on-campus locations.
  15. Shall be able to keep a wishlist, a list of items being sold that the user wishes to track.
- Moderators
  17. Shall be able to give an user a reason for why their item posting was denied.

## 3. UI Mockups/Storyboard (High Level)

### 3.1 Homepage

This is the first site the user will see when he opens our website. He can search and browse items as well as sign in and sell.

[Sell](#) | [Sign In](#)

# GatorGoods

Categories ▼

🔍 Search...

Title

Description ...

Contact Seller

Price

Title

Description ...

Contact Seller

Price

Title

Description ...

Contact Seller

Price

[About Us](#)

1 | 2 | 3

## 3.2 Post Item

User has to enter Title, category, price and description. If he wants to post an item without being logged in, the user must enter his credentials at the very end (lazy registration).

[Sell](#) | [Sign In](#)

# GatorGoods

Categories ▼

🔍 Search...

## Post Items

Title\*

Sofa

Catagory\*

Furniture ▼

Price\*

\$50.00

Description\*

Red Sofa

Upload an image (up to 5)\*

Upload

Cancel

Post

# GatorGoods

Categories ▼

🔍 Search...

Post

Title\*

Sofa

Catagor

Furnitu

Descrip

Red Sc

## Sign In Page

Email address

Email Address

Password

Password

Sign In

[Forgot password](#)

New to GatorGoods?

Register

Upload an image (up to 5)\*

Upload

Cancel

Post



### 3.3 Registration

Sell | [Sign In](#)

GatorGoods

Categories ▼ 🔍 Search...

Registration

Name \*

First

Last

Email Address \*

Password \*

Confirm Password \*

Captcha

☐ User Agreement [Link](#)

Register

### 3.4 Contact Seller

At this the potential buyer can contact the seller regarding the post.

[Sell](#) | [Sign In](#)

GatorGoods

Categories

▼

Q Search...

Math Book

Message

Hi, I would like to buy the math book. Please let me know if the location selected is okay and what time you are available to meet up.

Thanks.

Select Meetup Location

▼

[Cancel](#)

Send

Title

Description ...

Contact Seller

Price

[About Us](#)

1 | 2 | 3

## 3.5 Moderator

On this screen the moderator can approve and deny pending posts. He can also see a list of all users and is able to blacklist and reactive them, if necessary.

[Sell](#) | [Sign In](#)

# GatorGoods

Categories



Search...

## All Posts

▼ Post	▼ User	▼ Status	▼ Post Action
New Sofa	Sarah	Pending	Approve   Deny
Teaching Pointer	Montfermeil	Pending	Approve   Deny
Used Xbox One	John	Pending	Approve   Deny
Explosive Fireworks	Wes	Pending	Approve   Deny

## All Users

▼ Users	▼ Action
Bobby	Blacklist   Reactivate
John	Blacklist   Reactivate
Sarah	Blacklist   Reactivate
Stella	Blacklist   Reactivate

## 3.6 Dashboard

User can see all his active posts as well as all his messages.

[Sell](#) | [Sign In](#)

# GatorGoods

Categories ▼

🔍 Search...

### My Posts

▼ Post	▼ Status	▼ Action
New Sofa	Pending	Remove
Used Xbox One	Pending	Remove
Math Book	Approved	Remove
Explosive Fireworks	Denied	Remove

### My Messages

▼ Messages	▼ User
Hi I would like to purchase your sofa. Which brand ...	Sarah
Interested in the Xbox. Does it come with games?	Bob
Would you take \$40 for the book?	John
What is the condition of the sofa? Would you go lower...	Steve

## 4. High Level Architecture, Database Organization

### DB Organization:

items	
Field Name	Datatype
iid	ID
title	VARCHAR(60)
description	VARCHAR(500)
price	INTEGER
seller	Foreign Key (User)
category	Foreign Key (Category)
status	ENUM ('pending', 'approved', 'denied', 'sold')
createdAt	DATETIME
modifiedAt	DATETIME
approvedAt	DATETIME
isBook	BOOL
courseNumber	VARCHAR(10)

images	
Field Name	Datatype
imid	ID
item	Foreign Key (item)
imagePath	VARCHAR(4096)
thumbnailPath	VARCHAR(4096)

user_info	
Field Name	Datatype
uid	ID
firstName	VARCHAR(30)
lastName	VARCHAR(30)
moderator	BOOL
blacklisted	BOOL

**# We will use a user authentication library which will store email and password and will have a 1to1 relationship with the above table**

item_wishlist_user_junction	
Field Name	Datatype
item	Foreign Key (Item)
user	Foreign Key (User)

messages	
Field Name	Datatype
mid	ID
recipient	Foreign Key (User)
sender	Foreign Key (User)
sentAt	DATETIME
item	Foreign Key (Item)
content	VARCHAR(255)
location	Foreign Key (Location)

locations	
Field Name	Datatype
lid	ID
name	VARCHAR(50)
map	VARCHAR(4096)

categories	
Field Name	Datatype
cid	ID
name	VARCHAR(25)

#### Media Storage:

- Images: File System

#### Search/Filter Architecture and Implementation:

Algorithm/SW for search and category: %like

How will you organize search items for the user: Most similar result based on %like

What DB terms will be searched, how it will be coded and organized in the DB: Title, category, c

(and perhaps description, depending on performance)

#### List of APIs

##### 1. ItemAPI

Description: Sends all items of a certain user to the frontend. Differs between registered user and registered user. Just sends approved items if the requested user is not the same as the requesting registered user, otherwise will additionally send items with status of pending, denied, and sold. Does not include description.

##### 2. ItemDetailAPI

Description: Sends all of the details of an item including description, path to image, etc. If the requesting user is not a moderator or the owner of the object and the object's status is pending, denied or sold, the request should return a HTTP 403 error.

##### 3. BrowseItemAPI

Description:

- i. Selects items to show on the frontpage and as suggested items when no items could be found from a search result
- ii. Instead of passing via HTTP, this is based by setting a property on the window object for when this is used on the frontpage. For search results the procedure above will be used.

#### 4. MessageAPI

Description: Sends all messages of a certain user to the frontend

#### 5. WishlistAPI

Description: Sends all items of the wishlist of a certain user to the frontend

#### 6. AddWishlistItemAPI

Description: Creates a new wishlist item in the wishlist of a certain user

#### 7. DeleteWishlistItemAPI

Description: Deletes an item in the wishlist of a certain user

#### 8. CategoryAPI

Description: Sends all items of a certain category to frontend

#### 9. PendingItemsAPI

Description: Sends a list of items which are pending approval. Backend has to check if requesting user is moderator

#### 10. PostItemAPI

Description: Post item of a user. Includes approval by moderator.

#### 11. DeleteItemAPI

Description: Delete item of a certain user

#### 12. UpdateItemAPI

Description: Update item of certain user (description, price, title etc.)

#### 13. AddUserAPI

Description: Creates a new user based on the information the unregistered user made to get a registered user

#### 14. SearchAPI

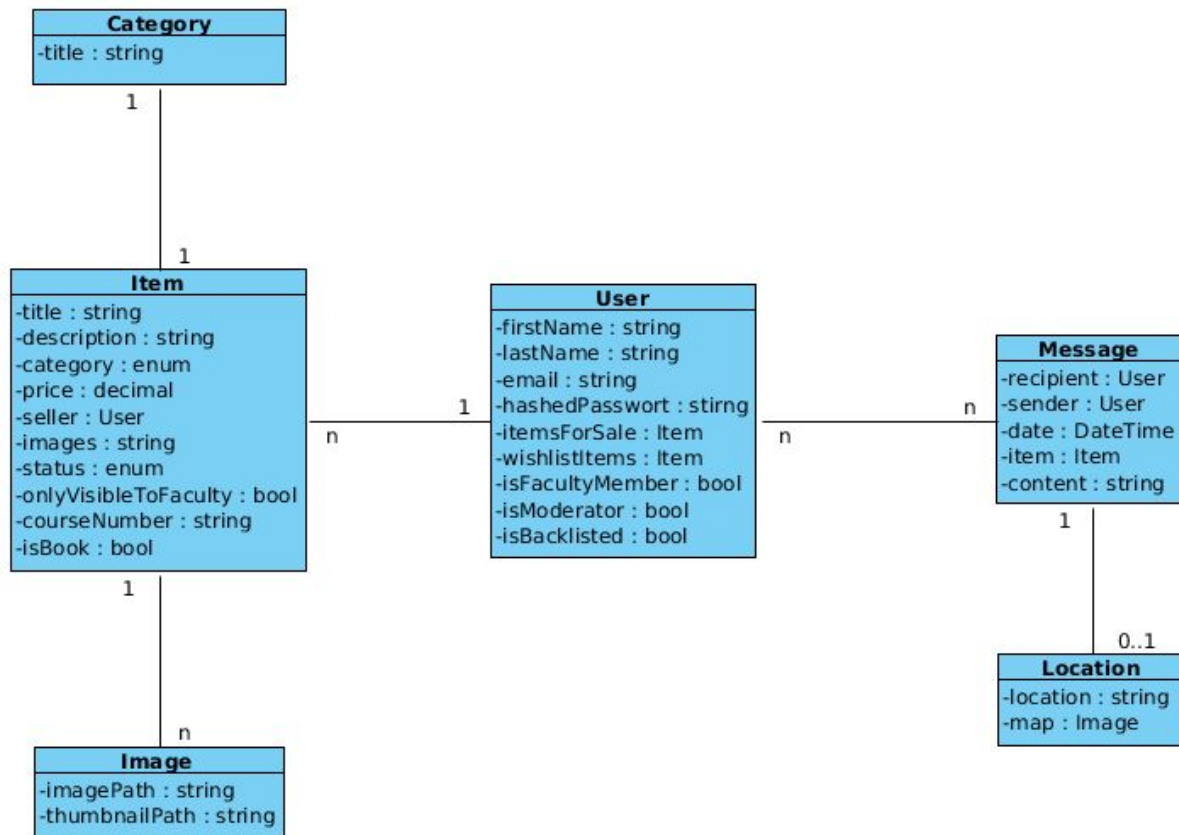
- a. Description



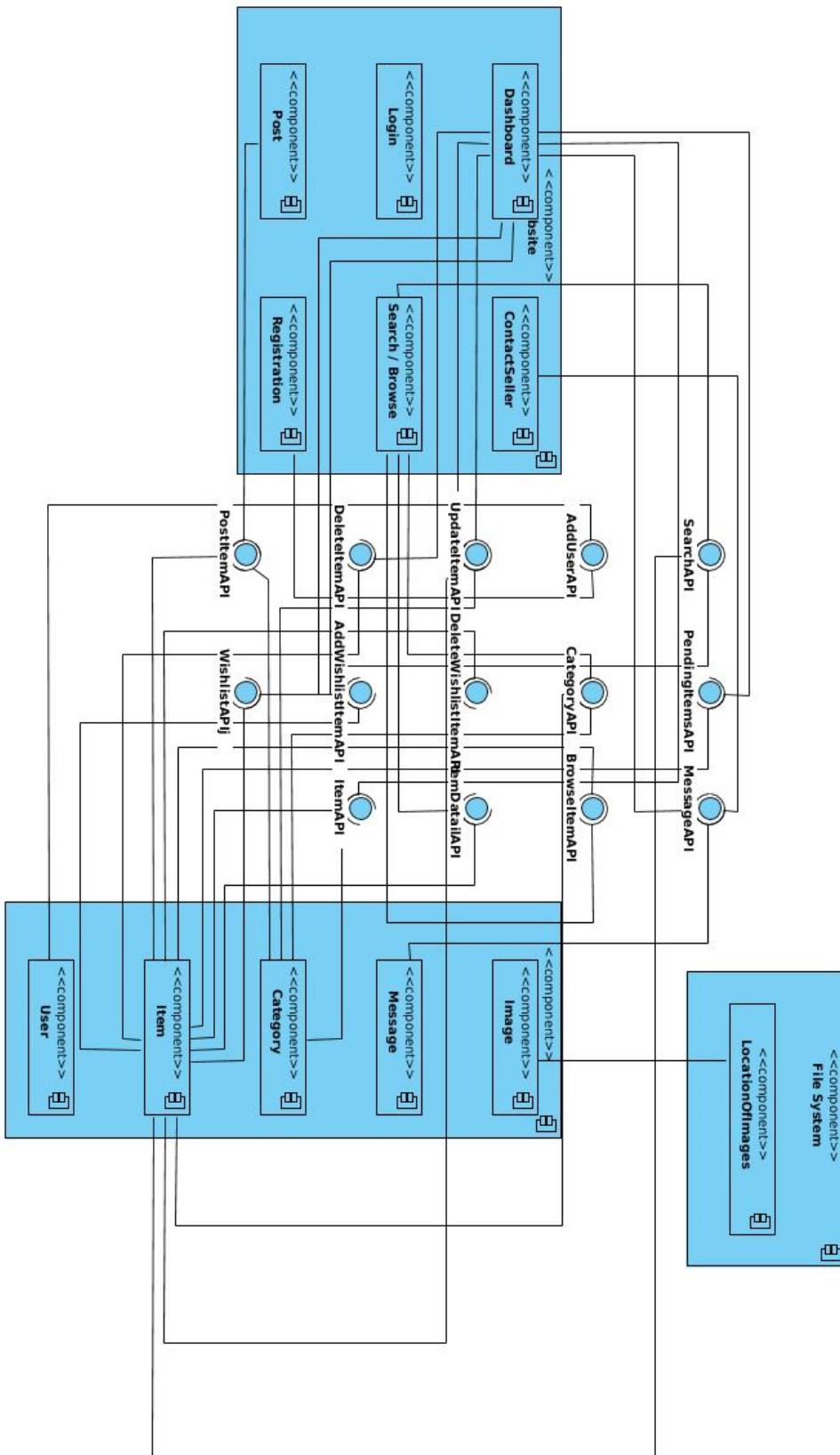
- i. Sends items to frontend based on search query provided in request
- b. Details
  - i. HTTP request
  - ii. Input: Search query
  - iii. Output: Ranked list of items JSON

## 5. High-level UML diagrams

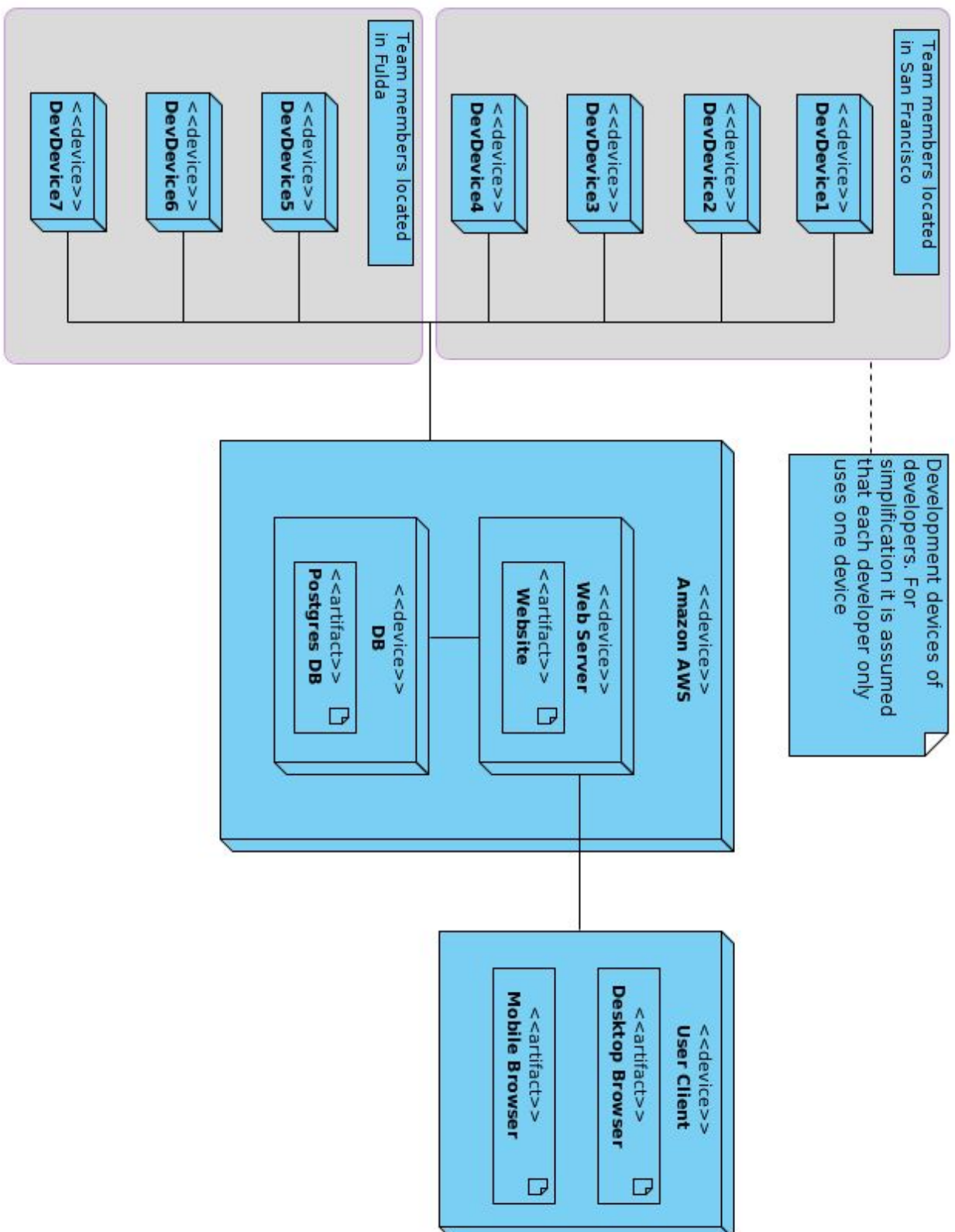
### 5.1 Class Diagram



## 5.2 Component Diagram



## 5.3 Deployment Diagram



## 6. Identify *actual* key risks for your project at this time

*Skill risks:* Lack of experience in frontend development at 3 out of 4 members

Member with experience teaches the others

*Schedule risks:* Vertical prototype

The vertical prototype seems to be very extensive and there can be a lot of unpredictable issues

*Teamwork risks:* a) Unsuitable project managing tool

b) Communication amongst locally distributed team

a) Check Trello, GitHub project managing and others

b) Clear communication, define precise APIs