* **Login**

The login will be based on the phone number. In principle, we can read the phone number from the device using a Cordova library. However, I read that this feature may be rejected by iOS.

In case the user is using another phone to login or in case of iOS, the user will be asked to enter his phone number in order to enter the service. We will store the phone number information in localStorage and use it automatically next time for automatic login.

In cases where we do not manage to read from the device and also fail to read from the localStorage, the user will be asked to enter his phone number again. We need to check if recent form filled data (the phone number in our case) is automatically offered to the users.

Note: At stage one we will read and create all the users registered in the google doc.

* **Login failure**

**Login may fail due to the following reasons**

1. **The user is not registered yet**
2. **The user is registered but fails to enter his phone number (missing numbers…)**
3. **Some communication/server error**

**In cases 1 and 2, we provide the user a feedback including the phone number. We will offer a registration screen including phone number and name. This will send a registration message to the administrator.**

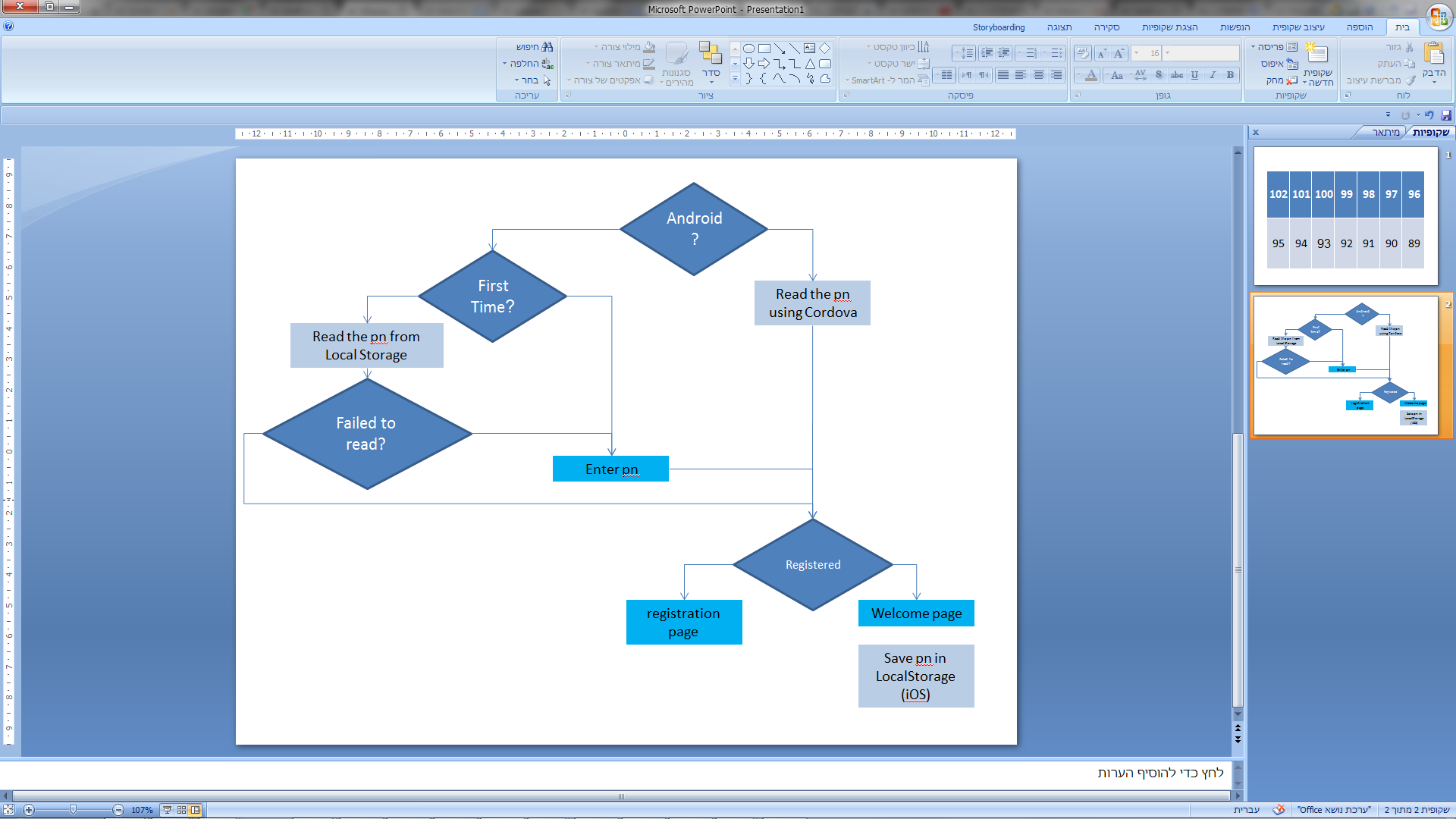
* **After the login**

After the login, the user should be presented with a welcome message with his name.

Should we retrieve all/most relevant data (personal information, future rides, preferences, history… ) on login?

* **Multiple users login from the same phone**

It may be possible that users will want to enter with some other user login, such as the case of helping a family member or friend. This is a possible but probably not a very common scenario. It will not be managed in stage one. A possible design is after the login to offer the user to login as another user by allowing him to feed the phone number.



Login flow diagram (right side is positive)

* **Testing login interface**

In order to be able to test the system, we need to be able to impersonate as different users. The testing phone numbers will be registered as devs. Upon login, devs will be redirected to a special login page were they can login as different users according to names or enter some phone number.