Use Case #1: Query Events for University

GENERAL CHARACTERISTICS

Author	Anirudh Bagde, Rupin Bhalla, Catherine Ta
Last Update:	9/24/16
Scope	Backend of HypeU event discovery service
Level	User level
Status	Under review
UML Diagram	See page 1 of ‘UML Diagrams.pdf’
Primary Actor	Client User/Guest: Someone looking for nearby events
Secondary Actors	Universities
Stakeholders and Interests	● Guest: Wants to find nearby events without having to log in. Might be visiting the university they are currently at, or someone living near a college campus. ● User: Wants to find nearby events by logging in. A user will be looking for events at his or her own college campus. ● University: Has the ability to monitor events, and possibly shut down malicious or inappropriate ones.
Preconditions	For guests: nothing For client users: already authenticated
Success Post Condition	All events at the current university are sent to the client
Failed Post Condition	An error status is sent to the client

MAIN SUCCESS SCENARIO (or basic flow)

Step	**Action**
1	User opens the client app, or goes to the discover tab
2	Client sends the user’s location and requests a list of nearby universities
3	Backend sends the client a list of nearby universities
4	Client requests all the events at the closest university
5	Backend sends events at the specific university to the client

EXTENSIONS or Alternate Flows

Step	**Branching Action**
4a	User selects a specific university they want to see events for 1. Client requests all the events at the user-specified university 2. Backend sends events at the specific university to the client
5a	Client requested events for a university that does not exist 1. Backend sends an error status to the client

SPECIAL REQUIREMENTS

Req Num	Requirement
1	Backend should gracefully handle bad queries (invalid location, nonexistent university, etc)

TECHNOLOGY AND DATA VARIATIONS LIST

Var Num	Variation
0	N/A

FREQUENCY OF OCCURRENCE: Very often, since this is the main functionality of the service

OTHER ISSUES

Issue Num	Issue
1	How many events to be returned; either by amount or date?