Project Early Bird

# Goal:

Create a service that will check various online ticket stores to detect the event tickets that are available to purchase. A mobile app should be developed to connect to this service and give users notifications and links to buy the ticket.

# Expectations:

- The service should crawl website as frequent as it can to give an early feedback for people to buy tickets before they are sold out.

- On mobile app users should see the events with related images and can easily subscribe to the event to get the notifications.

- E-mail, (opt.) SMS notifications should be used as well as mobile notifications.

- Users should select categories to filter the events

- Users should select geolocations to filter the events.

- (opt) Users can selects categories they like at first usage to prioritize these events when listing all events.

- (opt.) İntegration with 3rd party services (facebook,Spotify, etc.) to get the user preferences.

- The location of the user should be used to show the events nearby

- The mobile app should work on both Android and iOS devices.

- (opt.) The events that are currently available should be listed with the links to buy the tickets.

- An event might be flowed by 4-5k users. Total user number will be higher. The system should handle this load and could send email messages to that amount of users.

- (opt.) Facebook, Google single sign on login.

- (opt.) Share the events on whatsapp, Instagram, Facebook and Twitter.

- (opt.) Check the voice assistant API’s to check if an event ticket is available. (e.g. “OK Google, Is “Phantom of the Opera” tickets available to purchase [on ‘Early Bird’]”).

- (lamely opt.) Blink the Philips hue lights when the tickets are available :D

# Stakeholders:

* None for now. Content providers could be included for future extensions.

# Technologies considered:

* Python, Scrape for web crawling
* Xamarin, C sharp for mobile apps. (Cordova as an alternative)
* Firebase for DB and notification
* Email notification is not decided
* (opt.) Docker for containers, AWS for server side.
* Testing tools - not decided.
* Git for source control, distributed teams.

# Teams and Responsibilities:

Onur

Özge

# Risks:

* Main works of the team members may delay the project deadline.
* Aws or Scrapy may require some payment.
* Email notification may require payment.

# Quality Gateways:

* We need to have some :)

# Milestones:

* Project plan completed.
* 5 Main sites crawled
* 10 mile sites crawled
* DB, notification implemented
* Mockups and UI design of mobile app.
* Mobile App implemented.
* Alpha tests with close friends.
* Beta testing.
* Publishing on Play Store and Apple Store.
* Advertising on net by Instagram, Google and Facebook ads.