Software Architecture and Design

Assignment 4

Dr. Byron Williams

October 25th, 2016

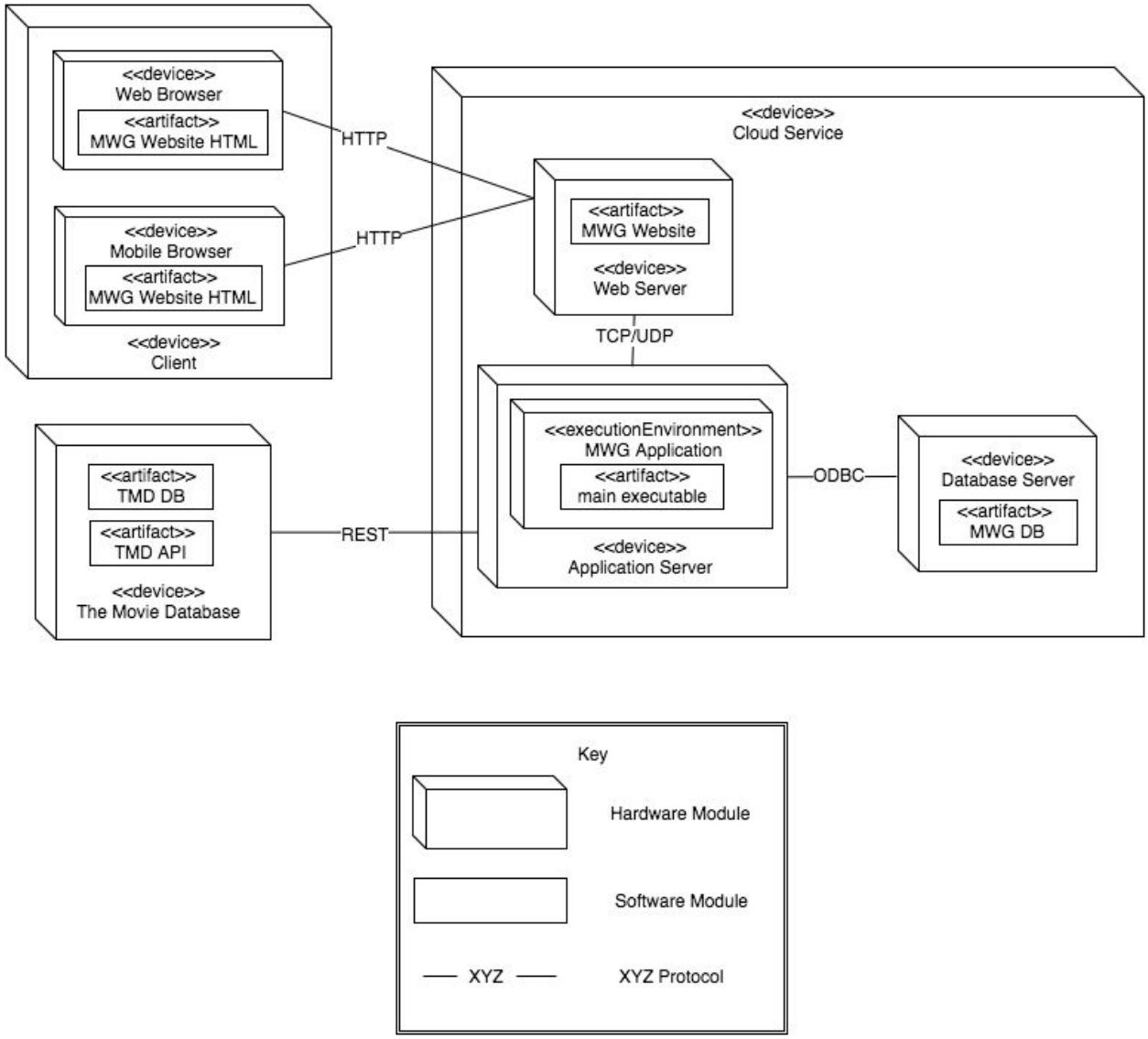
JJ Kemp (jhk114)
Hannah Church (hds109)

Lessons Learned

The combination of a REST API and JSON provided numerous learning opportunities. The organization of JSON made it quite simple to extract what data was necessary from each API return object. The formatting of REST in itself (GET, POST, DELETE, etc) made it simple to access data from the API. This also provides a simple way to implement The Movie Writers Guild's database in the future. Java was the programming language of choice because of level of familiarity and library availability. One of the libraries available to Java is Jackson, which provides a simplicity for JSON mapping.

Diagram Design

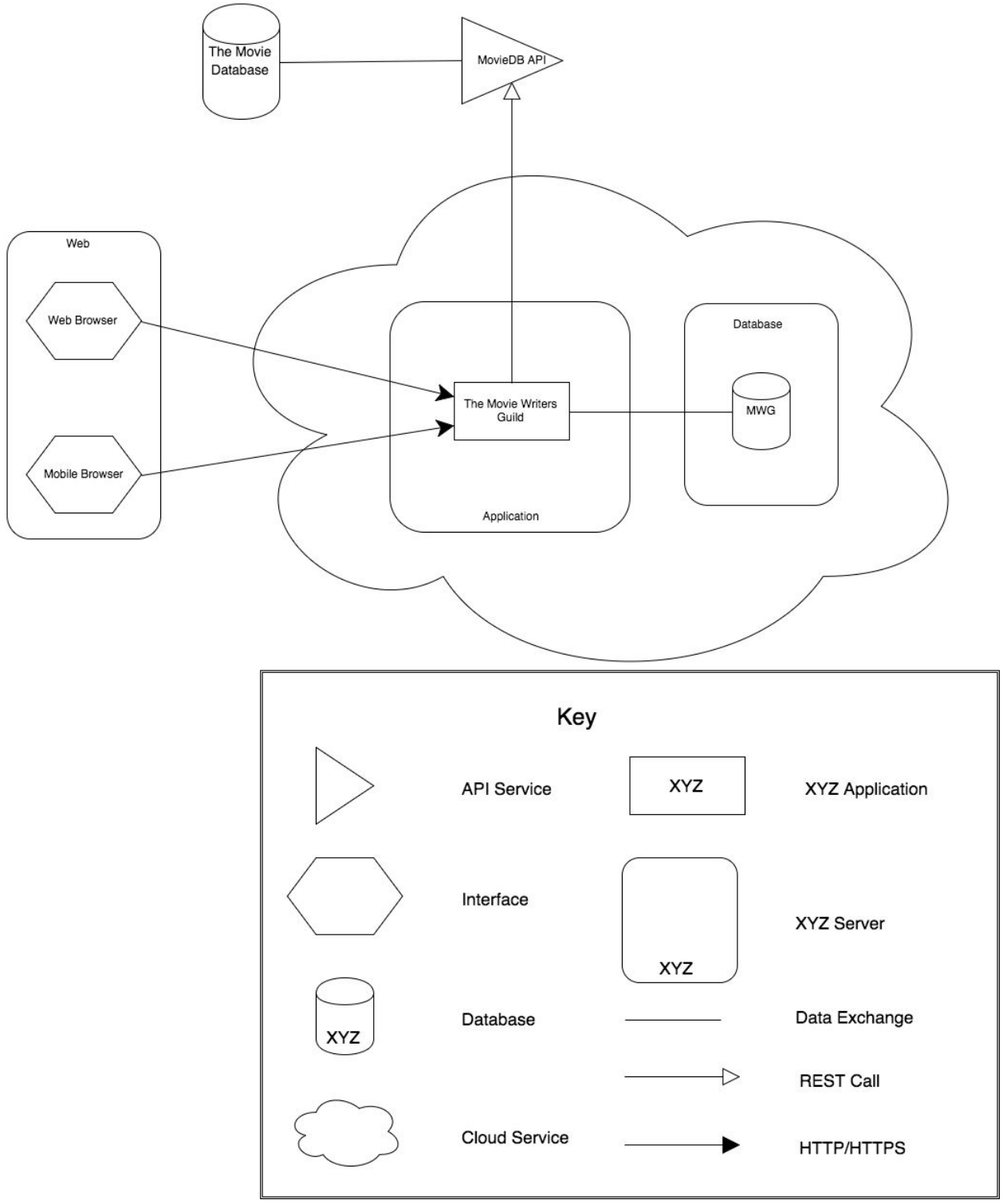
Deployment Diagram



The deployment diagram is a visual representation of how software is encompassed in the hardware of a system design. Hardware devices are denoted by 3-D boxes, and software artifacts are 2-D boxes. The cloud server hardware) includes the MWG application, database, and web servers. Within the application server is another device module, the execution environment. Within this environment is the main

executable behind the MWG application. The database server contains the MWG database, considered to be a software artifact. The database and application interact with one another through the open database connectivity (ODBC) protocol, which is a common protocol for application-database data interaction. The web server hosts the MWG website. The web server and application server communicate through a typical transmission control protocol (TCP). The web server also interacts with the client's device through simple hypertext transfer protocols (HTTP). The client's device is external to the cloud service, and encompases a mobile and desktop web browser. The browsers encompass the basic HTML of the MWG application's website. Another component, external to the cloud service, is The Movie Database application, which an API is used to connect to. In the deployment view these elements were left in a single TMD module, because they are not designed by the MWG team. The Movie Database application is interacted with through REST calls with the MWG application.

Client-Server C&C Diagram



Client-Server diagrams represent how services provide their services to clients. In contrast to the deployment diagram, the connector arrows are sometimes directional. The only connectors left unidirectional in this diagram are those that interact with a database, because data can flow in either direction. Similar to the deployment diagram, the application and database servers are hosted on the cloud service. Servers in

this diagram and denoted by rounded rectangles, and the cloud service is denoted by a cloud shape. The application communicates with the API via REST calls, and the API queries the TMD database for requested data. The MWG application also connects with the mobile and web browsers through HTTP requests, as in the deployment diagram.

Other REST APIs:

1) Twitter: https://dev.twitter.com/rest/public
    a) The SEARCH API: https://dev.twitter.com/rest/public/search
    b) Sample API call: https://api.twitter.com/1.1/search/tweets.json?q=%40twitterapi
        i) Searches tweets that reference the twitterAPI account
2) Google Glass: https://developers.google.com/glass/
    a) The Mirror API: https://developers.google.com/glass/v1/reference/
    b) Sample API call: https://www.googleapis.com/mirror/v1/timeline
        i) Gets a timeline
3) Flickr: https://www.flickr.com/
    a) Flickr API: https://www.flickr.com/services/api/
    b) Sample API call: http://api.flickr.com/services/rest/?&method=flickr.people.getPublicPhotos
        i) Gets a set of public photos
4) Amazon: http://docs.aws.amazon.com/AmazonS3/latest/API/Welcome.html
    a) S3 Storage API: http://docs.aws.amazon.com/AmazonS3/latest/API/Welcome.html
    b) Sample API call: http://s3-eu-west-1.amazonaws.com/mybucket/puppy.jpg
        i) Accesses puppy.jpg from a particular bucket
5) Tesla Model S: http://docs.timdorr.apiary.io/#
    a) API: http://docs.timdorr.apiary.io/#
    b) Sample API call: https://owner-api.teslamotors.com/api/1/vehicles/vehicle_id/wake_up
        i) ‘Wakes up’ the car connected to the API