Deployment Document

for

TimeTable Generator

Version 1.0

CS-08

Indian Institute of Information Technology Vadodara

Team Members

Aman Yadav (201651007)

DakshKumar Gondaliya (201651014)

Kirtika Singhal (201651024)

Mayank Pathela (201651029)

Nikhil Sachan (201651034)

Parmeshwar Kumawat (201651035)

Revision History

Version	Date	Name	Description
1	05/11/2018	Mayank Pathela	Initial Document

Table of Contents

1. About Heroku

Heroku is a cloud platform as a service (PaaS) supporting several programming languages. Heroku, one of the first cloud platforms, has been in development since June 2007. It supports Ruby, Java, Node.js, Scala, Clojure, Python, PHP, and Go.

2. Deploy

- The main content of the development are the source code, related dependencies if they exist, and a Procfile for the command.
- The application is sent to Heroku using either of the following: Git, GitHub, Dropbox, or via an API.
- There are packets which take the application along with all the dependencies, and the language runtime, and produce slugs. These are known as build-packs and are the means for the slug compilation process.
- A slug is a combination/bundle of the source code, built dependencies, the runtime, and compile / generated output of the build system which is ready for execution.
- Next is the Config vars which contain the customizable configuration data that can be changed independently of the source code.
- Add-ons are third party, specialized, value-added cloud services that can be easily attached to an application, extending its functionality.
- A release is a combination of a slug (the application), config vars and add-ons.
- Heroku maintains a log known as the append-only ledger of releases the developer makes.

3. Define the Application

- The definition of the application i.e. the source code and the description is built on the framework provided by Heroku which converts it into an application. The dependency mechanisms vary across languages: for Ruby the developer uses a Gemfile, in Python a requirements.txt, in Node.js a package.json, in Java a pom.xml, and so on.

4. Deploying Application

- The Application development on Heroku is primarily done through git. The application gets a new git remote typically named as Heroku along with its local git repository

where the application was made. Hence to deploy heroku application is similar to using the git push command.
- There are many other ways of deploying applications too. For example, developers can enable GitHub integration so that each new pull request is associated with its own new application, which enables all sorts of continuous integration scenarios. Dropbox Sync lets developers deploy the contents of Dropbox folders to Heroku, or the Heroku API can be used to build and release apps.
- Deployment then, is about moving the application from a local system to Heroku.

5. Steps to be Followed to deploy the application:-

- Change environment keys from development to production.
- Navigate to Project folder on terminal and follow these:-
  - Run “git init” on terminal
  - Run “git add .” on terminal
  - Run “git commit” on terminal
  - Run “heroku Login” on terminal
  - Run “heroku create” on terminal
  - Add Production Database keys in our application.
  - Set heroku remote branch
  - Push all the commits
  - Set front-end build script in root package.json file

6. Link Generated:-

https://afternoon-springs-33996.herokuapp.com/

6. GitHub Repository Link:-

https://github.com/starkblaze01/Jenereta/