

SAN FRANCISCO PLANNING DEPARTMENT

Executive Summary Abbreviated Institutional Master Plan

HEARING DATE: JULY 21, 2016

| Date: | June 29, 2016 |
|------------------|--|
| Case No.: | 2016-004046IMP |
| Project Address: | 98 Battery Street |
| Zoning: | C-3-O (Downtown Office) District |
| | 450-S Height and Bulk District |
| | Downtown Plan Area |
| Block/Lot: | 0266/008 |
| Project Sponsor: | Sylvain Kalache |
| | Holberton School |
| | 98 Battery Street, Suite 402 |
| | San Francisco, CA 9411 |
| Staff Contact: | Nicholas Foster – (415) 575-9167 |
| | nicholas.foster@sfgov.org |
| Recommendation: | No action necessary – informational item |

1650 Mission St. Suite 400 San Francisco, CA 94103-2479

Reception: 415.558.6378

Fax: 415.558.6409

Planning Information: 415.558.6377

BACKGROUND ON INSTITUTIONAL MASTER PLANS

Holberton School (the "School" or the "Project Sponsor") has submitted an Abbreviated Institutional Master Plan ("IMP" or the "Plan") for consideration by the Planning Commission ("Commission") as required by Section 304.5 of the Planning Code.

Planning Code Section 304.5 requires post-secondary educational and medical institutions in the city to provide the Planning Commission with a long-range development plan every 10 years, with updates provided every two years. Institutions located in the C-3 (Downtown, Commercial) Districts occupying, or proposing to occupy, less than 100,000 square feet of floor area may submit an Abbreviated IMP. An Abbreviated IMP requires a description of the following: the institution's physical plant; employment characteristics; services provided and service population; ownership of properties throughout the City and County of San Francisco; the impact on the "Eight Priority Policies" of Planning Code Section 101.1; and parking availability and other relevant general information.

The purpose of the IMP is to provide this information to the Commission and the public. The IMP is available for public review, and has been posted on the Planning Department's website. Any proposed changes in land use described in an IMP would require separate review and approval by the Commission and/or department staff, as applicable.

PROJECT DESCRIPTION

The project is the submission of an Abbreviated IMP for Holberton School. This Abbreviated IMP represents the School's first submission an IMP.

SUMMARY OF THE CONTENTS OF THE ABBREVIATED IMP

Founded in May of 2015, Holberton School is a small, private software engineering (programming) school based in Downtown San Francisco. The School offers a two-year (four-semester) "full-stack" software engineering program ("Program"), where students learn a variety software programming skills. Students are expected to complete a total of 2,160 hours of instruction as well as a six-month internship as part of the Program. The School promotes both project-based and peer-learning learning techniques as a means of teaching software engineering.

The School is currently seeking accreditation from the Western Association of Schools and Colleges (WASC) and the State of California's Bureau of Private Postsecondary Education.

Population Characteristics:

Currently, the School enrolls approximately 32 full-time students in its four-semester program. The School anticipates future enrollment to range between 32 and 42 full-time students. The School maintains a staff of three (3) full-time instructors and may hire additional staff/faculty as the School's needs dictate.

Facilities:

Holberton School does not currently own any property in the City and County of San Francisco for the purposes of academic or administrative uses. At present, the School's physical plant is located at 98 Battery Street, Suite 402. This building is located on the east side of Battery Street, between Pine and Bush Streets. The School began leasing approximately 4,026 square feet of office space on the 4th floor of the subject building in October of 2015. This space is currently being used by the School for academic instruction purposes. The conversion of the existing office space on the 4th floor constituted a change of use from Office to Institutional Use (Institutional Educational Use), which, is principally permitted within the C-3-O Zoning District.

As the School occupies less than 100,000 square feet in the C-3 Zoning District, it is therefore eligible to submit an Abbreviated IMP.

Parking and Campus Access:

The School does not provide any off-street parking for its leased space at 98 Battery Street. The Plan states the School's parking philosophy is to encourage public transportation and carpooling for its students, staff, and faculty. The School's proximity to Market Street will undoubtedly afford users access to an array of transit options. The 98 Battery Street property is located immediately adjacent to the Embarcadero Street BART and MUNI Metro Station, as well as numerous Market Street MUNI bus and streetcar lines. In addition, the subject property is located within walking distance of the temporary Transbay Terminal—as well as the proposed new facility—and the Ferry Building, providing access to an array of regional transit options.

Current Projects/Future Expansion:

The Plan does not call for any immediate expansion of facilities beyond basic tenant improvements to the leasable space on the 4th floor of 98 Battery Street (e.g. interior renovations and facility upgrades to support the School's educational and operational requirements).

Any future development plans would be described in an update to the Planning Commission, in the form of an Update to the Abbreviated Institutional Master Plan. Any projects proposed by the School would require standard Planning Department review upon their submission; receipt of this Plan does not convey approval of any or all proposed projects within it.

ENVIRONMENTAL REVIEW STATUS

The Project is categorically exempt from the California Environmental Quality Act ("CEQA"), Section 15301 (Class 1, Minor Alternations to Existing Structures).

PUBLIC COMMENT

No public comment has been received by the Department since the filing of the application.

REQUIRED COMMISSION ACTION

The item is an informational item, and no action is required. However, the Planning Commission has the discretion under Planning Code Section 304.5(d) to hold or not hold public hearing on an Abbreviated IMP. If the Commission requests a hearing, it will be scheduled for a later date.

STAFF RECOMMENDATION

The Holberton School Abbreviated IMP includes all information required by Planning Code Section 304.5. The staff recommendation is to accept the Abbreviated IMP as submitted and not hold a hearing.

RECOMMENDATION: Informational Only; Do not require a public hearing on this IMP

Attachments: Block Book Map Sanborn Map Zoning Map Context Photographs Holberton School Abbreviated IMP Floor Plans (98 Battery Street; 4th Floor)

Exhibits

SAN FRANCISCO PLANNING DEPARTMENT

Parcel Map



SAN FRANCISCO PLANNING DEPARTMENT

Sanborn Map*



*The Sanborn Maps in San Francisco have not been updated since 1998, and this map may not accurately reflect existing conditions.

Zoning Map





Aerial Photo





Site Photo



Street View of 98 Battery Street.





ABBREVIATED INSTITUTIONAL MASTER PLAN

Holberton School 98 Battery Street – suite 402 San Francisco, CA 94111

Submitted Date: May 2016 Approval Date: _____

HOLBERTON SCHOOL

Abbreviated Institutional Master Plan

Submitted May 2016

Prepared for THE CITY OF SAN FRANCISCO Planning Commission

Owner HOLBERTON SCHOOL 98 Battery Street – suite 402 San Francisco, CA – 94111

Owner Contact

Julien BARBIER CEO & Co-founder 415.358.0819

Prepared by

Sophie RIGAULT-BABIER Accountant/HR/Office Manager 415.358.0819 sophie@holbertonschool.com

TABLE OF CONTENTS

| 1. | Intro | p - 4 |
|----|---------------------------------|--------|
| 2. | Programs and Mission | p - 5 |
| 3. | Curriculum | p - 6 |
| 4. | Population | p - 19 |
| | a/ Students | |
| | b/ Staff | |
| | c/ Affirmative Action | |
| 5. | Tuition | p - 20 |
| 6. | Facilities | p - 21 |
| | a/ Street Map | |
| | b/ Floor Map | |
| | c/ School photos | |
| 7. | Parking & Public Transportation | p - 24 |
| 8. | Licensing and Accreditation | p - 25 |
| 9. | Expansion Plans | p - 26 |

1. INTRODUCTION

This Abbreviated Institutional Master Plan (this "*AIMP*") is submitted on behalf of Holberton, Inc., d/b/a Holberton School ("*Holberton*" or the "*School*"), pursuant to the City of San Francisco Planning Code, Section 304.5, which permits an postsecondary educational institution presently occupying or proposing to occupy a site area of less than 50,000 square feet or 100,000 square feet in the C-3 District, and which places on file with the San Francisco Planning Department a statement that the institution does not anticipate any future expansion to more than 50,000 square feet or 100,000 square feet in the C-3 District, to file an abbreviated institutional master plan.

Holberton presently occupies less than 50,000 square feet and does not anticipate any future expansion to more than 50,000 square feet. The information in this AIMP is accurate as of the date set forth below.

2. PROGRAMS and MISSION

Holberton offers a two-year higher-education program, which includes an internship of six months between the two years. Students who complete the program earn a full stack software engineer diploma. The goal of Holberton is not to teach students a specific programming language or framework, but to teach problem solving in order to better prepare the students for challenges faced during their careers. Students will learn the tools needed to accomplish objectives, including low-level and system programming, higher level programming, web and mobile development, system administration and operations, open-source, algorithms, reverse engineering, documentation, communication, and community building.

Using project-based learning and peer learning, Holberton's mission is to train the best software engineers of their generation.

- <u>Project-Based Learning</u>. Project-based learning provides students with increasingly difficult programming challenges, while giving them minimal initial instruction regarding how to solve them. As a consequence, students naturally seek out the theory and tools they need to solve the challenges, and work collaboratively to complete the projects. The project-based learning approach pragmatically approximates the challenges software engineers face in the real world. As a result, Holberton students are better prepared to work in the tech industry, and prepared to learn on the fly, ensuring their adaptability through the quickly-evolving technological landscape.
- <u>Peer Learning</u>. Peer learning is an educational practice in which students interact with other students to attain educational goals. Coupled with project-based learning, it allows Holberton students to unleash their creativity and naturally learn how to work as a team to solve challenges. At Holberton, most projects are collaborative. Holberton encourages students to share their knowledge and help each other. When a student successfully explains a concept to another student, everyone wins. The recipient will likely better understand the concepts, as studies have shown that abstract concepts are better understood when explained by peers. And the student performing the explanation will better achieve knowledge consolidation. Peer education fosters a very constructive learning culture, as students are immersed into an environment where everyone is driven to help each other.

3. CURRICULUM

Holberton's Full-Stack Software Engineering Program (the "*Program*") is a two-year program where students are expected to complete a total of 2,160 hours of instruction.

The curriculum is articulated around the following core skills teaching courses taught in classrooms using online resources, books and videos:

- 1. Low-level programming
- 2. High-level programming
- 3. Web development
- 4. System administration
- 5. Software industry soft skills

| Name of Course | Description & Objectives | Study Time |
|-------------------------------|---|------------|
| | Module Description | |
| Low level programming: | In this module students learn the very basics of | 90 hours |
| Introduction to UNIX and low- | UNIX and low-level programming. | |
| level programming | Learning Objectives | |
| | At the completion of this module students will | |
| | know | |
| | - Work on a UNIX-like operating system, | |
| | understand and manipulate the user environment | |
| | and file system | |
| | - The basics of C-programming (functions, loops, | |
| | variables, conditions, pointers, data structures, | |
| | linked lists, function pointers) | |
| | | |
| | Module Description | 146 hours |
| Low level programming: UNIX | In this module students learn how to manipulate | |
| programming | the POSIX API, perfect their C-programming | |
| | skills and learn UNIX multitasking. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know | |
| | - UNIX I/O | |
| | - How to work with processes and jobs | |
| | - Inter-processes communication (pipes signals) | |
| | - UNIX UI (termcaps) | |
| | | |

| Name of Course | Description & Objectives | Study Time |
|---|--|------------|
| | | |
| | | |
| Low level programming: C Programming - Basics | Module Description In this module students learn about algorithms and complexity. | 140 hours |
| | At the completion of this module students will know: | |
| | how to and when to use data structures (tables, hash tables, trees, graphs) standard algorithms (search, sort) | |
| Low level programming: Security - Basics | Module DescriptionIn this module students learn about the basics ofdigital securityLearning ObjectivesAt the completion of this module students willunderstand / know:- why security is important- basics of cryptographyhasian of ouditing of C source and basics | 50 hours |
| | basics of auditing a C source code how to alter a binary to execute malicious code | |
| Low level programming: Advanced UNIX programming | Module DescriptionIn this module students learn more about UNIXsystem, and networkingLearning ObjectivesAt the completion of this module students willknow:- Internal UNIX structure | 140 hours |
| | Network programming (sockets, select, protocols) Thread, mutex, semaphore | |
| Low level programming: Assembly | Module DescriptionIn this module students learn about X86Assembly programming and how to reverseengineer basic programs.Learning ObjectivesAt the completion of this module students will | 80 hours |

| Name of Course | Description & Objectives | Study Time |
|------------------------------|---|------------|
| | know: | |
| | - Asm X86 | |
| | - Basics of reverse engineering | |
| | Module Description | 20 hours |
| Low level programming: Shell | In this module students learn about shell scripting | |
| scripting | Learning Objectives | |
| | At the completion of this module students will | |
| | know | |
| | - Sh Shell scripting | |
| | - Bash Shell scripting | |
| | | |
| | Module Description | 20 hours |
| Web dev: Basic front-end | In this module students learn about HTML5. | |
| technologies | CSS3, and introductory JavaScript for the | |
| | browser (DOM, the window object, the document | |
| | object. etc.) | |
| | Learning Objectives | |
| | At the completion of this module students will be | |
| | able to build full webpages easily from a design | |
| | or from scratch, and build basic dynamism on ton | |
| | of it | |
| | 01 11. | |
| | Module Description | 16 hours |
| Web dev: Basic back-end | In this module students learn to change the | ro nours |
| technologies | information displayed depending on context | |
| | using the PHP language: they will also learn how | |
| | to get information from user input using the PHP | |
| | language. The projects involved revolve around | |
| | building a CRUD scaffolding to manage users | |
| | and a simple micro blogging platform | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | At the completion of this module students will | |
| | tasky algorized interest with each other, and will | |
| | technologies interact with each other, and will | |
| | know now to build a basic full-stack website. | |
| | Madula Description | 20 hours |
| Web dev: Advanced front-end | In this we dole sty dents have a base to be for the | 20 nours |
| technologies | In this module students learn about the future | |
| gg | evolutions of USS (USS4) and JavaScript | |
| | (ECMA/). They will learn to use productivity | |
| | tools (such as the Sass pre-processor, JQuery, and | |

| Name of Course | Description & Objectives | Study Time |
|----------------------------------|--|------------|
| | an introduction to CoffeeScript). They will also learn about web quality (accessibility, SEO, | |
| | performance), and how to ensure it. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how to code (non-single-page) front-end | |
| | applications like in the industry, efficiently and | |
| | with high quality. | |
| | Module Description | 20 hours |
| Higher-level dev: advanced | In this module students learn about the advanced | 20 110015 |
| JavaScript as a language | features and syntax of JavaScript in general (not | |
| | just in the browser), and will do some projects | |
| | they did before in C language, but this time in | |
| | JavaScript. They will also learn of all the various | |
| | ways JavaScript code gets executed (in a | |
| | browser, on the command line, in a back-end, in a | |
| | task manager like Gulp or Grunt, interpreted or | |
| | compiled, etc.). | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | start to understand the differences between | |
| | languages based on the C / Javascript example, | |
| | and the ways code in general gets executed in all | |
| | languages. | |
| | Module Description | 40 hours |
| Higher-level dev: an overview of | In this module students learn an overview about | |
| languages | all of the relevant programming languages | |
| | currently used in the industry (JavaScript, Ruby, | |
| | PHP, Hack, Java, Clojure, Python, Go, Scala, | |
| | Perl, Swift, Objective C, C++, etc.). Projects will | |
| | revolve around using at least one interpreted | |
| | script language (Ruby), one compiled type-safe | |
| | language (Java), and JavaScript, that they already | |
| | know in depth. They will have to experiment with | |
| | at least one of the other ones. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | tully understand the global difference between all | |
| | industry-relevant languages, and the interest in | |

| Name of Course | Description & Objectives | Study Time |
|---------------------------------|--|------------|
| | having so many. They will be able to make | |
| | decisions about which technology makes most | |
| | sense for a given project or task. | |
| | Module Description | 40 hours |
| Higher-level dev: advanced data | In this module students learn about the data | |
| structures | structures that they haven't already discovered in | |
| | C language, and that higher-end languages | |
| | usually come with, such as dynamic arrays, sets, | |
| | hash tables (dictionaries), etc. or more exotic | |
| | ones, such as the "Queues" or "StringBuffers" in | |
| | the Java language. The students will also learn | |
| | about which languages offer closure and why, as | |
| | well as more exotic concepts such as | |
| | Procs/lambda/blocks in Ruby. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how to decide which data structure is best | |
| | for a given data they have to pro cess, and the | |
| | difference between native data structures across | |
| | the programming languages landscape. | |
| | | |
| | Module Description | 30 hours |
| Higher-level dev: performance | In this module students learn about computing | |
| | performance, both theoretical (big-O | |
| | assessments) and empirical (benchmarks). | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | not only know how to ensure continuous great | |
| | performance for their projects, but also the | |
| | traditional performance pitfalls. | |
| | | |
| | Module Description | 40 hours |
| Higher-level dev: ensuring code | In this module students learn about the various | |
| quality | code quality assessment tools (such as Radar), | |
| | and proper code review and pull request | |
| | processes. They will also be introduced to | |
| | automated testing (unit testing, functional testing, | |
| |) and the Test-Driven Development (TDD) | |
| | approach. | |
| | Learning Objectives | |
| | At the completion of this module students will | |

| Name of Course | Description & Objectives | Study Time |
|-----------------------------------|--|------------|
| | know how to maintain the quality of a project | |
| | through time. | |
| | | |
| II's have have here a feature | Module Description | 50 hours |
| Higher-level dev: software | In this module students learn about RESTful | |
| architecture | APIs and microservices, design patterns, and | |
| | mutualization of code (library, packages, etc.) as | |
| | it's done across modern languages. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know the choices they have to make when | |
| | designing an architecture, both inside an | |
| | application, and in how applications talk to each | |
| | other. They will also know how to reuse code, but | |
| | also how to publish and share their own code. | |
| | _ | |
| | Module Description | 40 hours |
| Higher-level dev: object-oriented | In this module students learn about object- | |
| programming | oriented programming and all of its usual | |
| | concepts; they will use that knowledge in | |
| | building an Object Relational Mapping tool | |
| | (ORM). | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how objects work across modern | |
| | languages, and be comfortable using them for | |
| | advanced needs. | |
| | | |
| | Module Description | 40 hours |
| Higher-level dev: software | In this module students learn about external | |
| security | attacks against software. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how to protect themselves against external | |
| | attacks against software. | |
| | | |
| | Module Description | 30 hours |
| Higher-level dev: advanced SQL | In this module students learn advanced usage of | |
| and advanced source control | both the SQL language to use relational databases | |
| | (joins, inner queries, indexes,) and advanced | |
| | usage of Git as well of other source control tools | |

| Name of Course | Description & Objectives | Study Time |
|------------------------------------|---|------------|
| | (SVN, Mercurial,) | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how to compose complex SQL queries, and | |
| | optimize them; and they'll be familiar with all the | |
| | current ways to version source code. | |
| | | |
| | Module Description | 50 hours |
| Higher-level dev: advanced | In this module students learn about regular | |
| language features | expressions (using them with the Perl language). | |
| | reflection/meta-programming (using it with | |
| | Ruby) non-blocking I/O and promises (using it | |
| | with JavaScript) functional programming (using | |
| | it with Scala) | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know about the most state of the art technical | |
| | nor diama that they might not use every day in | |
| | their fature ich bet will unlock correles | |
| | their future job, but will unlock complex | |
| | situations on the day the need arises. | |
| | Modulo Description | 50 hours |
| Higher-level dev: advanced data | In this module students learn shout no SOI | JU HOUIS |
| tools | In this module students learn about no-SQL | |
| | databases (document-oriented databases, and key- | |
| | value storage), RAM storage of data (pros and | |
| | cons), and webservice protocols other than REST | |
| | (such as SOAP, or technology-dependent ones | |
| | such as JMS). | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how the make the best decision about how | |
| | to store and transit data, depending on the context | |
| | and the data itself. | |
| | | |
| | Module Description | 60 hours |
| Higner-level dev: state-of-the art | In this module students learn about introductory | |
| specializations | artificial intelligence, machine learning, data | |
| | science and data visualization. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | have an entry-level knowledge of the state-of-the- | |

| Name of Course | Description & Objectives | Study Time |
|------------------------------|---|------------|
| | art fine specializations that are currently very | |
| | sought after in the industry. | |
| | | |
| Mobile: Android development | Module Description | 60 hours |
| Wobie. And ou development | In this module students learn about how an | |
| | Android application gets developed, based on | |
| | their knowledge of the Java language. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how to build and ship an Android | |
| | application, and will own one of their own on the | |
| | Google Play Store. | |
| | Madula Description | 60 hours |
| Mobile: iOS development | In this module students learn shout how on iOS | ou nours |
| • | application gots developed, while learning the | |
| | Objection gets developed, while learning the | |
| | Transk framework | |
| | Lour Iramework. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how to build and ship an iOS application, | |
| | and will own one of their own on Apple's App | |
| | Store. | |
| | Module Description | 35 hours |
| System administration: Cloud | In this module students work with the most | |
| | common Cloud provider such as Amazon AWS. | |
| | Google cloud engine, Gandi VPS. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know the specificity, advantage and | |
| | disadvantages of each Cloud provider and will be | |
| | able to use their services | |
| | | |
| System administration: | Module Description | 65 hours |
| operating system | In this module students interact with server-side | |
| | operating systems by setting up, configuring and | |
| | maintaining them. They will learn by working | |
| | with Bash and basic system commands. | |
| | Learning Objectives | |
| | At the completion of this module students will | |

| Name of Course | Description & Objectives | Study Time |
|---------------------------------|---|------------|
| | understand and be able to manage, debug server- | |
| | side operating systems. | |
| | | |
| | Module Description | 55 hours |
| System administration: CI/CD | In this module students learn about the CI/CD | |
| (continuous integration and | concept and why is it so widely used in the | |
| continuous deployment) | industry. They will build a CI/CD infrastructure | |
| | from scratch that will feature a one click deploy | |
| | capability, testing code via uni and integration | |
| | tests, ship the code and measure performance | |
| | impact. | |
| | Learning Objectives | |
| | At the completion of this module students will be | |
| | able to interact and build company level CI/CD | |
| | systems. | |
| | | |
| | Module Description | 48 hours |
| System administration: | In this module students learn about the | |
| configuration management | configuration management concept, why more | |
| | and more system administrators in the industry | |
| | use it and what are the differences between them. | |
| | They will work with the most popular ones like | |
| | Puppet, Ansible or Docker. | |
| | Learning Objectives | |
| | At the completion of this module students will be | |
| | able to use any type of configuration management | |
| | tool to configure system infrastructures. | |
| | | |
| | Module Description | 55 hours |
| System administration: database | In this module students learn about databases. | |
| | differences between the 2 mains types: relational | |
| | and key-value store. The will setup, configure | |
| | and manage multiple of these databases and | |
| | create industry level infrastructure such a slave | |
| | and read-only setures | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | have a solid understand of how databases work | |
| | and will be able to setup and interact with them | |
| | and will be able to setup and interact with them. | |
| | | |
| | | |

| Name of Course | Description & Objectives | Study Time |
|------------------------------|--|------------|
| | Module Description | 40 hours |
| System administration: DNS | In this module students learn about how DNS | |
| | (Domain Name System) works by ordering their | |
| | own domain name, installation and configure a | |
| | DNS server, hosting their domain name/zone on | |
| | it and finally build a web interface to administrate | |
| | it. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | understand how DNS works and why we use it. | |
| | They will be able to work with domain names to | |
| | make their web applications or services easily | |
| | accessible. | |
| | | |
| | Module Description | 28 hours |
| System administration: | In this module students learn about the | |
| Documentation | importance of documentation and will write | |
| | monitoring alert documentation as well as system | |
| | infrastructure documentation. | |
| | Learning Objectives | |
| | At the completion of this module students will be | |
| | able to write any type of documentation that is | |
| | required in the software industry and will be | |
| | trained to document everything that should be | |
| | documented. | |
| | | |
| | Module Description | 50 hours |
| System administration: basic | In this module students learn about the basis of | |
| networking | networking: protocols | |
| | TCP/UDP/HTTP/HTTPS/ICMP/ARP, IPv4, | |
| | addressing private/public, NATing, sockets, basic | |
| | routing and subnetting. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | have the basic knowledge to understand how | |
| | networks are working. They will be able to setup | |
| | basic networking infrastructure and perform | |
| | operation on existing ones. These skills will be | |
| | very useful in system architecture, security and | |
| | operating system domains. | |
| | | |

| Name of Course Description & Objectives | | Study Time |
|---|---|------------|
| Sustain a durinistication account | Module Description | 50 hours |
| System administration: security | In this module students are educated about | |
| | general security knowledge: man in the middle, | |
| | keylogger, social engineering, targeting email. | |
| | They will also use SSH keys to connect to unix | |
| | system, perform network sniffing, scan their | |
| | network and system infrastructures to find | |
| | security breach and finally setup and manage | |
| | firewalls. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | have a solid package about how to protect their | |
| | infrastructure against hackers and they will have | |
| | basic knowledge about how computer and system | |
| | attacks are done. | |
| | | 45.1 |
| System administration: system | Module Description | 45 hours |
| architecture | In this module students learn about concepts to | |
| | infractructures. They will implement: | |
| | asymphronous systems, redundant/load balanced | |
| | asynchronous systems, redundant/load balanced | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know and have implemented all classical system | |
| | architectures that you can find in the software | |
| | industry | |
| | Module Description | 58 hours |
| System administration: | In this module students learn to monitor systems | 00 nours |
| monitoring | such as websites or API services by installing | |
| | configuring monitoring tools as well as | |
| | responding to them. | |
| | Learning Objectives | |
| | At the completion of this module students will | |
| | know how to setup the right monitoring to ensure | |
| | that their websites and services are up and | |
| | running properly and will be able to setup rules to | |
| | get alerted if not. | |
| | | |
| | | |
| | | |

| Name of Course Description & Objectives | | Study Time | | |
|---|---|------------|--|--|
| | Module Description | 56 hours | | |
| System administration: | In this module students learn about virtualization, | | | |
| virtualization | how it is working, what are the advantage | | | |
| | compared to bare metal machines. They will also | | | |
| | use virtualization and work with the latest | | | |
| | containerization technologies. | | | |
| | Learning Objectives | | | |
| | At the completion of this module students will be | | | |
| | able to use all virtualization and containerization | | | |
| | technologies such as VirtualBox or Docker. | | | |
| | Module Description | | | |
| System administration: web | In this module students learn about the different | | | |
| stack | layers of a web stack: web servers, caching | | | |
| | servers, queuing servers. They will install and | | | |
| | configure them to host the websites and services | | | |
| | they coded. They will also work on debugging | | | |
| | exercises. | | | |
| | Learning Objectives | | | |
| | At the completion of this module students will be | | | |
| | able to setup and use any type of web stack that | | | |
| | will be serving production traffic. | | | |
| | Module Description | 81 hours | | |
| Soft skills | In this module students practice soft skills that | | | |
| | are required to be a great software engineer. | | | |
| | Learning Objectives | | | |
| | At the completion of this module, students will | | | |
| | have practiced soft skills that can make the | | | |
| | difference between a good and a great software | | | |
| | engineer. Students will be able to communicate | | | |
| | efficiently, verbally and by writing, in front of | | | |
| | one person or in front of a crowd. They will be | | | |
| | able to manage a project using different | | | |
| | methodologies like Scrum or waterfall. | | | |
| | Module Description | 81 hours | | |
| Entrepreneurship | In this module students are introduced to | | | |
| | entrepreneurship and be practicing skills that | | | |
| | relate to it such as performing a market analysis, | | | |
| | building a business plan, pitching or project or | | | |

| Name of Course | Description & Objectives | Study Time |
|----------------|---|------------|
| | raising money. | |
| | Learning Objectives | |
| | At the completion of this module students will have the basis they need to bootstrap their own startup. | |

Each course is taught using specific tasks and projects. Students are evaluated on theoretical, practical, research and communication skills, and graded on a pass/fail basis. There are periodic evaluations as well as a final examination at the end of each year. Students must maintain an 80% average or better of passing grades to maintain satisfactory academic status and qualify for a completion certificate.

After the first year, students will have to complete a six month internship with a company in the technology sector before being allowed to start the second year of the Program. The goal of the internship is to enable the student to apply the knowledge and skills acquired during the school year to a practical context. Students will be graded by their internship mentor, based on the following criteria:

- Leadership skills;
- Execution skills; and
- Craftsmanship skills.

The Program is available to students with or without any prior training in computer science or programming experience. There are no prerequisites to enrollment. The teaching methodology is centered exclusively around project-based learning and peer education.

4. POPULATION

a/ <u>Student</u>

Holberton School is open to all individuals who possess a High School diploma or equivalent. The inaugural class begins on January 22nd, 2016 with 32 full-time students. Thereafter, it is expected that the School will enroll between 32 and 42 full-time students in its four-semester program. Students are expected to come from throughout the Bay Area and the United States, as well as Canada and several other countries.

b/ Staff and Faculty

The School will initially employ three full time instructors, and may possibly hire more staff/faculty members in the future, as the School's needs dictate.

Holberton is and will be at all times committed to complying fully with the Americans with Disabilities Act and ensuring equal opportunity in employment for qualified persons with disabilities. All employment practices and activities are and will be conducted on a nondiscriminatory basis.

c/ Affirmative Action

Holberton does not discriminate in employment opportunities or practices on the basis of race, color, religion, creed, gender, sexual orientation, marital status, age, national origin, ancestry, veteran's status, disability, medical condition, or any other basis that is protected by law.

5. TUITION

The first incoming class will be entirely tuition free. Thereafter, tuition fees for the twoyear Program will be \$45,000.

6. FACILITIES

Holberton currently leases one campus location at 98 Battery Street #402, San Francisco, CA 94111, which is located in the C-3-O Downtown district. Holberton rents this space from 98 Battery Associates, LLC, a Nevada limited liability company, the building owner. The School occupies approximately 4,026 square feet on the fourth floor of the building.

Holberton does not own any real property in the City or County of San Francisco and has no current plans to acquire any real property in the City or County of San Francisco.

The School does not, and has no current plans to, provide student housing or student housing assistance to its students.



a/ <u>STREET MAP</u>

b/ FLOOR MAP

- Lobby
- Open Space
- Conference Space
- 3 offices
- Kitchen



c/ <u>SCHOOL PHOTOS</u>

LOBBY



CONFERENCE SPACE



OPEN SPACE

KITCHEN



6. PARKING & PUBLIC TRANSPORTATION

There is no off-street parking provided by Holberton School; however, the 345 California Center Garage, which is located at 345 California Street, has an entrance located on Battery Street and a capacity of 180 vehicles. The School has no plans to provide any off-street parking.

Holberton encourages the use of public transportation and carpools for students and employees with similar schedules. Due to its central location, Holberton is at the hub of several Bay Area transportation systems. 98 Battery Street is located 1 block from the Embarcadero and Montgomery BART Stations. Additionally, 98 Battery Street is also served by MUNI Bus Routes.

7. LICENSING and ACCREDITATION

Holberton School is currently in the process of applying for an approval to operate from the Bureau for Private Postsecondary and Vocational Education in Sacramento, California. The School intends to begin the accreditation application process in 2016.

8. EXPANSION PLANS

While Holberton does not currently anticipate to expand its facilities or operations in San Francisco County, its intent is to grow nationally and internationally in the near to distant future.

Respectfully submitted on January 12, 2016

HOLBERTON, INC. d/b/a HOLBERTON SCHOOL

6 By: /

Julien BARBIER, CEO



ž CAD\SK 5057

| CUPANCY TYPE | AREA | LOAD FACTOR | OCC. LOAD |
|-----------------|-------------|----------------|--------------|
| TION | 238.57 SF | 100 SF | 3 |
| OFFICE | 2,022.12 SF | 100 SF | 20 |
| | 430.62 SF | 100 SF | 5 |
| NG | 327.68 SF | 20 SF | 16 |
| N | 123.00 SF | 200 SF | 1 |
| | | | 45 |



| SCALE: | 1/8" = 1'-0" | SHEET |
|------------|--------------|-------|
| DATE: | 8.13.15 | |
| PROJ. NO.: | 15057 | |

r number

 $\leq |\langle 1 |$