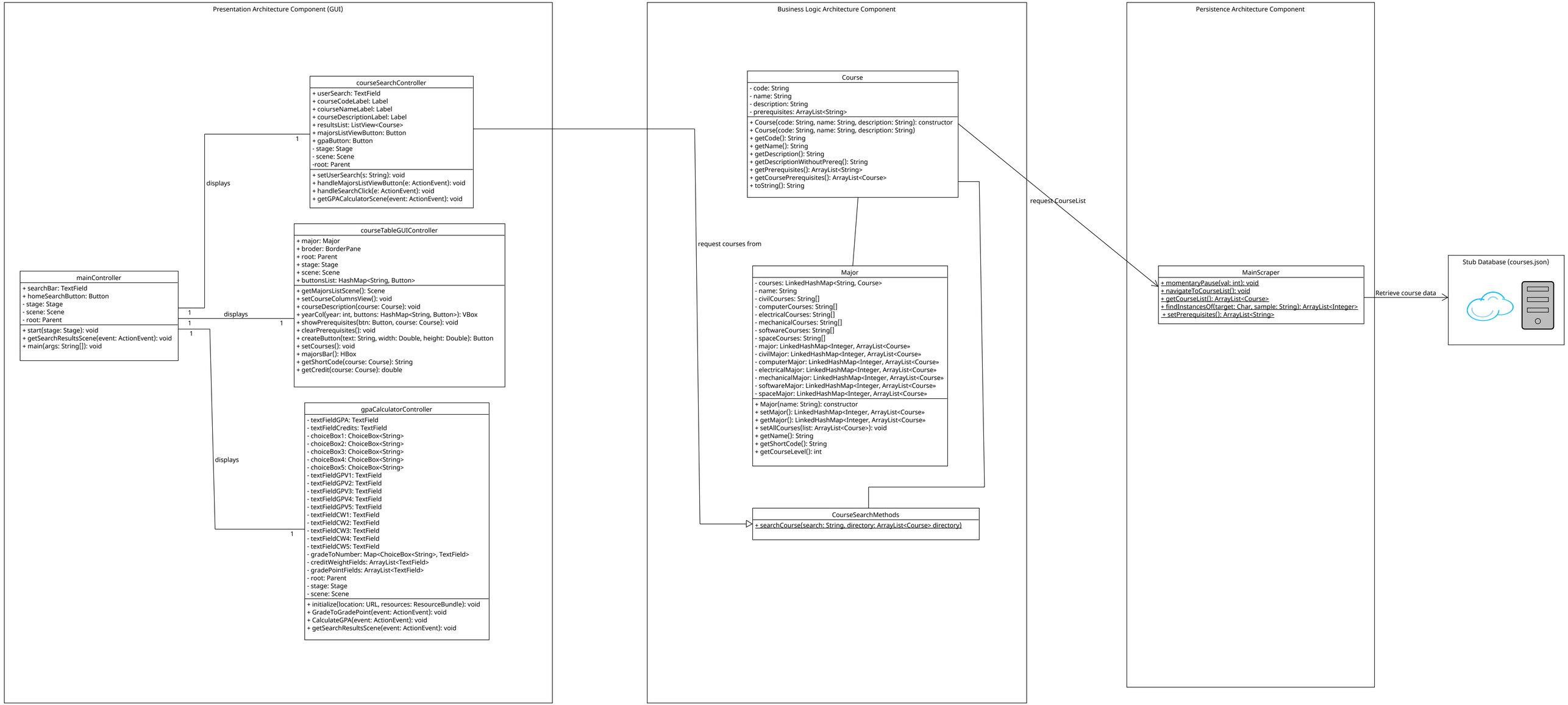
Presentation Architecture Component (GUI)
courseSearchController
+ userSearch: TextField
+ courseCodeLabel: Label
+ coiurseNameLabel: Label
+ courseDescriptionLabel: Label
+ resultsList: ListView<Course>
+ majorsListViewButton: Button
+ gpaButton: Button
- stage: Stage
- scene: Scene
-root: Parent
+ setUserSearch(s: String): void
+ handleMajorsListViewButton(e: ActionEvent): void
+ handleSearchClick(e: ActionEvent): void
+ getGPACalculatorScene(event: ActionEvent): void
courseTableGUIController
+ major: Major
+ broder: BorderPane
+ root: Parent
+ stage: Stage
+ scene: Scene
+ buttonsList: HashMap<String, Button>
+ getMajorsListScene(): Scene
+ setCourseColumnsView(): void
+ courseDescription(course: Course): void
+ yearCol(year: int, buttons: HashMap<String, Button>): VBox
+ showPrerequisites(btn: Button, course: Course): void
+ clearPrerequisites(): void
+ createButton(text: String, width: Double, height: Double): Button
+ setCourses(): void
+ majorsBar(): HBox
+ getShortCode(course: Course): String
+ getCredit(course: Course): double
mainController
+ searchBar: TextField
+ homeSearchButton: Button
- stage: Stage
- scene: Scene
- root: Parent
+ start(stage: Stage): void
+ getSearchResultsScene(event: ActionEvent): void
+ main(args: String[]): void
displays
1
displays
1
1
1
displays
1
gpaCalculatorController
- textFieldGPA: TextField
- textFieldCredits: TextField
- choiceBox1: ChoiceBox<String>
- choiceBox2: ChoiceBox<String>
- choiceBox3: ChoiceBox<String>
- choiceBox4: ChoiceBox<String>
- choiceBox5: ChoiceBox<String>
- textFieldGPV1: TextField
- textFieldGPV2: TextField
- textFieldGPV3: TextField
- textFieldGPV4: TextField
- textFieldGPV5: TextField
- textFieldCW1: TextField
- textFieldCW2: TextField
- textFieldCW3: TextField
- textFieldCW4: TextField
- textFieldCW5: TextField
- gradeToNumber: Map<ChoiceBox<String>, TextField>
- creditWeightFields: ArrayList<TextField>
- gradePointFields: ArrayList<TextField>
- root: Parent
- stage: Stage
- scene: Scene
+ initialize(location: URL, resources: ResourceBundle): void
+ GradeToGradePoint(event: ActionEvent): void
+ CalculateGPA(event: ActionEvent): void
+ getSearchResultsScene(event: ActionEvent): void
Business Logic Architecture Component
Course
- code: String
- name: String
- description: String
- prerequisites: ArrayList<String>
+ Course(code: String, name: String, description: String): constructor
+ Course(code: String, name: String, description: String)
+ getCode(): String
+ getName(): String
+ getDescription(): String
+ getDescriptionWithoutPrereq(): String
+ getPrerequisites(): ArrayList<String>
+ getCoursePrerequisites(): ArrayList<Course>
+ toString(): String
request courses from
Major
- courses: LinkedHashMap<String, Course>
- name: String
- civilCourses: String[]
- computerCourses: String[]
- electricalCourses: String[]
- mechanicalCourses: String[]
- softwareCourses: String[]
- spaceCourses: String[]
- major: LinkedHashMap<Integer, ArrayList<Course»
- civilMajor: LinkedHashMap<Integer, ArrayList<Course»
- computerMajor: LinkedHashMap<Integer, ArrayList<Course»
- electricalMajor: LinkedHashMap<Integer, ArrayList<Course»
- mechanicalMajor: LinkedHashMap<Integer, ArrayList<Course»
- softwareMajor: LinkedHashMap<Integer, ArrayList<Course»
- spaceMajor: LinkedHashMap<Integer, ArrayList<Course»
+ Major(name: String): constructor
+ setMajor(): LinkedHashMap<Integer, ArrayList<Course»
+ getMajor(): LinkedHashMap<Integer, ArrayList<Course»
+ setAllCourses(list: ArrayList<Course>): void
+ getName(): String
+ getShortCode(): String
+ getCourseLevel(): int
CourseSearchMethods
+ searchCourse(search: String, directory: ArrayList<Course> directory)
request CourseList
Persistence Architecture Component
MainScraper
+ momentaryPause(val: int): void
+ navigateToCourseList(): void
+ getCourseList(): ArrayList<Course>
+ findInstancesOf(target: Char, sample: String): ArrayList<Integer>
+ setPrerequisites(): ArrayList<String>
Retrieve course data
Stub Database (courses.json)