OpenMRS is hoping to be a mentoring organization for Google Summer of Code™ 2023! Since 2007, we’ve enjoyed participating in this great program and we’re extremely excited about the projects and mentorship opportunities available this year. Coding for OpenMRS is a great way to practice your coding skills and, at the same time, help benefit people in developing countries who are on the front lines of the battle against HIV/AIDS, TB, Malaria, and other public health challenges.

If you are new to OpenMRS, we recommend starting with our Guide to the New & Curious. It will introduce you to our community, the tools and spaces we use, and help you get to know the different squads and teams working on various community projects.

For a more detailed history of who we are and what we do, please see here.

---

GSoC 2023 Program Administrators

Jennifer Antilla
Kakumirizi Daud
Jayasanka Weerasinghe

Please see GSoC Admin Guidelines for more information.

Selected Projects for GSoC 2023

<table>
<thead>
<tr>
<th>Project Name &amp; Outcome</th>
<th>Project Rating and Length</th>
<th>About the project</th>
<th>Required skills</th>
<th>Selected Contributor</th>
<th>Primary Mentor</th>
<th>Backup Mentor</th>
</tr>
</thead>
<tbody>
<tr>
<td>O3: Print feature</td>
<td>Medium 175 hours</td>
<td>&quot;Print this&quot; component in O3 that can be used/configured depending on what the user wants to print. E.g. med list, recent labs, etc. Will need to do review of what people are printing / where in EMR they need/have the print button these days. Some implementers need specific areas of the chart to have a print option; others use general record printouts, e.g. a summary the patient can take with them to other locations in lieu of a digital medical record integration.</td>
<td>React</td>
<td>deborah Kalungi</td>
<td>Hadijah Kymepeire</td>
<td>Dennis Kigen</td>
</tr>
<tr>
<td>Responsiveness &amp; Tablet-based Needs for OpenMRS 3</td>
<td>Medium 175 hours</td>
<td>We have a ton of designs for 3.x that show how the frontend UX should change based on screen size or device (e.g. large desktop, small laptop, or tablet). Much of this responsiveness still needs to be implemented consistently throughout the application. Watch this video for more information.</td>
<td>React Javascript Typescript</td>
<td>JESSIE SSEBULIBA</td>
<td>Vineet Sharma</td>
<td>Kumuditha Karunarathna</td>
</tr>
<tr>
<td>Improving the OpenMRS Developer Experience: Updating the SDK</td>
<td>Advanced 350 hours</td>
<td>It's time to clean up our SDK and improve the developer experience.</td>
<td>Java</td>
<td>Wikum Weerakutti</td>
<td>Daniel Kayiwa</td>
<td>Joshua Nserek</td>
</tr>
</tbody>
</table>
### Potential GSoC contributors discuss application ideas with mentoring organizations

Potential GSoC contributors discuss application ideas with mentoring organizations. The tests will be set up to achieve a high level of test coverage. Unit and integration tests of OpenMRS3, aimed at achieving a consistent and comprehensive test strategy or framework for the different developers. To address this issue, there is a need for a consistent and comprehensive test strategy or framework for the unit and integration tests of OpenMRS3, aimed at achieving a high level of test coverage.

### Program Timeline

Look at [here](#) for more info on the full GSoC 2023 program timeline.

<table>
<thead>
<tr>
<th>Project</th>
<th>Complexity</th>
<th>Duration</th>
<th>Description</th>
<th>Mentors</th>
<th>Developer</th>
</tr>
</thead>
<tbody>
<tr>
<td>O3: Migrate vanilla React forms to FHIR</td>
<td>Medium</td>
<td>175 hours</td>
<td>We currently have a bunch of forms written in O3 in vanilla React. These forms, though functional, are not the most performant or extensible and for the most part, lack validation. React Hook Form offers easy-to-use validation out of the box, and more coverage is expected. The goal of the project is to refactor existing forms to use the React Hook Form library as well as Zod for schema validations.</td>
<td>Christopher Lumu</td>
<td>Shubham Sharma</td>
</tr>
<tr>
<td>O3: Extending E2E Automated Tests for the OpenMRS3.0 RefApp</td>
<td>Medium</td>
<td>175 hours</td>
<td>At present, there are already a few E2E tests running for the openmrs-esm-patient-management repository and more coverage is expected. The goal of the project is for someone to extend the E2E tests for openmrs-esm-patient-chart and openmrs-esm-core using the playwright framework. The tests will be set up to automatically validate the user interface based on specific workflows.</td>
<td>Randilla Premaratne</td>
<td>Heshan Andrews</td>
</tr>
<tr>
<td>O3: Draw on a Body Diagram</td>
<td>Medium</td>
<td>175 hours</td>
<td>OpenMRS is a medical record system that has previously allowed the uploading and visualisation of patient conditions through diagrams. This feature has been critical in helping doctors better explain patient conditions to their patients. However, the module lacked certain essential functionalities that could improve the user experience. Therefore, this project aims to improve the diagramming feature in OpenMRS by allowing the upload of any diagram as an image, annotation of certain areas of the diagram with different shapes, saving and retrieving these diagrams.</td>
<td>React, TypeScript</td>
<td>Jayasanka Weerasinghe</td>
</tr>
<tr>
<td>OpenMRS Android Client: Add more features from the web app</td>
<td>Advanced</td>
<td>350 hours</td>
<td>The potential goals for this year could be (but are not limited to): 1. Add more features from the reference application to the Android client which are as follows: 1.1) Add and schedule patient Appointments 1.2) Add the Vitals of a patient 1.3) Add and view Orders of a patient 1.4) Add and view Observations of a patient 2. Increase the test coverage of the app (due from previous GSoC Project) 3. Migrate remaining Java classes of the code to Kotlin</td>
<td>Android, Kotlin, Java, Unit Testing, Android Architecture</td>
<td>Namanya Abert</td>
</tr>
<tr>
<td>FHIR: Add support for FHIR PATCH operations</td>
<td>Medium</td>
<td>175 hours</td>
<td>OpenMRS is using the FHIR API more and more in place of the REST API. However, the FHIR API is, by default, quite verbose. Supporting PATCH operations would allow us to support partial updates to FHIR resources without needing to send the whole resource from the client to the server.</td>
<td>Java</td>
<td>Ian Bacher</td>
</tr>
<tr>
<td>O3: Enhancing Component Test Coverage and Quality for OpenMRS3</td>
<td>Medium</td>
<td>175 hours</td>
<td>OpenMRS consists of several micro frontend modules and repositories, most of which have unit and integration tests written by individual developers. However, these tests exhibit inconsistencies due to the diverse testing strategies used by different developers. To address this issue, there is a need for a consistent and comprehensive test strategy or framework for the unit and integration tests of OpenMRS3, aimed at achieving a high level of test coverage.</td>
<td>React, TypeScript</td>
<td>Ayush Mishra, Piumal Rathnayake</td>
</tr>
</tbody>
</table>

### GSoC 2023 preparations

- **January 23**: Mentoring organizations can begin submitting applications to Google
- **February 7**: Mentoring organization application deadline
- **February 22**: List of accepted mentoring organizations announced
- **February 22 - March 19**: Potential GSoC contributors discuss application ideas with mentoring organizations
- **March 20**: GSoC contributor application period begins
- **April 4**: GSoC contributor application deadline
- **April 27**: GSoC contributor proposal rankings due from Org Admins
- **May 4**: Accepted GSoC contributor projects announced
- **May 4 - 28**: Community Bonding Period. Students get to know mentors, read documentation, prepare for work on their projects
- **May 28**: Coding officially begins
- **May 29 - August 21**: Coding Period for Medium project
- **May 29 - November 6**: Coding Period for Advanced project
- **July 10**: Mentors and GSoC contributors can begin submitting midterm evaluations
- **July 14**: Midterm evaluation deadline (standard coding period)
- **July 14 - August 21**: Work Period | GSoC contributors work on their project with guidance from Mentors
- **August 21 - 28**: Final week | GSoC contributors submit their final work product and their final mentor evaluation (coding period)
- **August 28 - September 4**: Mentors submit final GSoC contributor evaluations (standard coding period) for medium size project.
- **September 5**: Initial results of Google Summer of Code 2023 announced
- **September 4 - November 6**: GSoC contributors with extended timelines continue coding
- **November 6**: Final date for all GSoC contributors to submit their final work product and final evaluation
- **November 13**: Final date for mentors to submit evaluations for GSoC contributor projects with extended deadlines

### Guidelines

- **Student's guidelines**
  - GSoC - Guidelines for Students
  - GSoC - Proposal Guidelines
- **Mentor's guidelines**
  - GSoC - Guidelines for Mentors
  - GSoC - Project Creation Guidelines

### OpenMRS resources to know

- GitHub: [https://github.com/openmrs](https://github.com/openmrs)
- Talk Forum: [https://talk.openmrs.org](https://talk.openmrs.org)
- Help Desk: [https://help.openmrs.org](https://help.openmrs.org)
- Issue Tracker (JIRA): [https://issues.openmrs.org](https://issues.openmrs.org)
- Wiki: [https://wiki.openmrs.org](https://wiki.openmrs.org)